

Shape Calculus: Timed Operational Semantics and Well-formedness

Ezio Bartocci¹ Diletta Romana Cacciagrano¹
Maria Rita Di Berardini¹ Emanuela Merelli¹ Luca Tesei¹

Abstract

The Shape Calculus is a bio-inspired calculus for describing 3D shapes moving in a space. A shape forms a 3D process when combined with a behaviour. Behaviours are specified with a timed CCS-like process algebra using a notion of channel that models naturally binding sites on the surface of shapes. Processes can represent molecules or other mobile objects and can be part of networks of processes that move simultaneously and interact in a given geometrical space. The calculus embeds collision detection and response, binding of compatible 3D processes and splitting of previously established bonds. In this work the full formal timed operational semantics of the calculus is provided, together with examples that illustrate the use of the calculus in a well-known biological scenario. Moreover, a result of well-formedness about the evolution of a given network of well-formed 3D processes is proved.

1 Introduction

In a near future, systems biology will profoundly affect health care and medical science. One aim is to design and test “in-silico” drugs giving rise to individualized medicines that will take into account physiology and genetic profiles [14]. The advantages of performing in-silico experiments by simulating a model, instead of arranging expensive in-vivo or in-vitro experiments, are evident. But of course the models should be as faithful as possible to the *real system*.

Towards the improving of the faithfulness of languages and models proposed in literature in the field of systems biology, the Shape Calculus [6, 5, 4]

¹School of Science and Technology, University of Camerino
Via Madonna delle Carceri 9, 62032, Camerino (MC), Italy.
Email: `name.surname@unicam.it`

was proposed as a very rich language to describe mainly, but not only, biological phenomena. The main new characteristics of this calculus are that it is spatial - with a geometric notion of a 3D space - and it is shape-based, i.e. entities have geometric simple or complex shapes that are related to their “functions”, that is to say, in the context of formal calculi, the possible interactions that can occur among the entities (called 3D processes, in the same context). Note that there are some other approaches that use a geometric notion of space [2, 16, 3], but they do not fully exploit the potentiality of geometry: they only consider positions that are centers of perception/communication spheres or discretized the space using grids. Moreover, differently to classical mathematical models for biological systems, the Shape Calculus is individual-based, that is to say it considers autonomous entities that interact with others in order to give rise to an emerging behaviour of the whole system. However, this characteristic is also present in other languages and models proposed in literature [18, 19, 11, 7, 12]. Parallel to the introduction of the Shape Calculus, a simulation environment called BIOSHAPE [20, 9] has been proposed. This environment embodies all the concepts and features of the Shape Calculus, plus others, more detailed, characteristics that are related to the implementation of simulations in different biological case studies [9, 8, 10]. The Shape Calculus can be considered the formal core of BIOSHAPE. On the Shape Calculus, formal verification techniques can be applied, while in-silico experiments (simulations) can be performed in BIOSHAPE. These two approaches can be used successfully in an independent way, but they can also be combined to address complex biological case studies.

In [5], the language of the Shape Calculus was introduced with the main aim of gently and incrementally present all its features and their relative semantics. The motivations behind the type and nature of the calculus operators were discussed and a great variety of scenarios in which the calculus may be used effectively were described. In [4] the timed operational semantics of the calculus was introduced and a well-formedness property of the dynamics of the calculus was proved.

This paper is an extended version of [4] in which the original concepts of the calculus are fully recalled and in which further examples are provided in order to explain better the technical details of its timed operational semantics.

We present the full timed operational semantics of the Shape Calculus and we explain with proper running examples all the technical points that

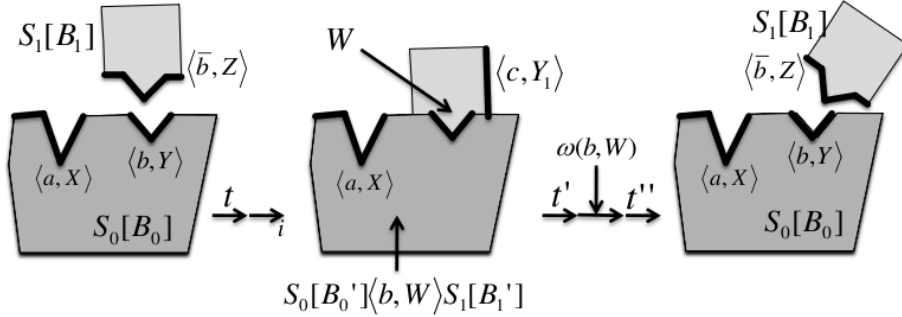


Figure 1: An example of binding and subsequent weak-split of two 3D processes.

need particular attention and further explanation. Moreover, we define a concept of well-formedness, starting from shapes and porting it to more complex calculus objects such as 3D processes and, ultimately, networks of 3D processes. In the calculus, only well-formed objects are considered. Well-formedness is a standard concept used to avoid strange and unwanted situations in which a term can be legally written by syntax rules, but that semantically corresponds to a contradictory situation, for instance, in our case, a composed shape whose pieces move in different directions. We prove that a given well-formed network of 3D processes always evolves into a well-formed network of 3D processes, that is to say, no temporal and spatial inconsistencies are introduced by the dynamics of the calculus.

The paper is organized as follows. Section 2 recalls the main concepts of the calculus and focuses on weak and strong split operations whose semantics are slightly more complicated to define than that of other operators. Section 3 introduces formally 3D shapes, shape composition, movement, collision detection and collision response. Section 4 defines behaviours and 3D processes giving them full semantics. Section 5 puts all the pieces together and specifies precisely networks of 3D processes and a general result of dynamic well-formedness is proven. Finally, Section 6 concludes with ongoing and future work directions. For the sake of readability all the proofs have been moved in Appendices A, B and C.

2 Recall of Main Concepts

In this section we recall the main concepts the calculus introduced in [5] and we focus in particular to the weak and strong split operations, in order to make them clear from the beginning and, thus, then smoothly define their semantics, which requires some particular technical expedients.

The general idea of the Shape Calculus is to consider a 3D space in which several shapes reside, move and interact. While time flows, shapes move according to their velocities, that can change over time both due to a general motion law - for instance as in a gravitational or in an electromagnetic field, or as in Brownian motion - and due to collisions that can occur between two or more shapes. Collisions can result in a bounce, that is to say *elastic* collisions. Instead, as it often happens in biological scenarios, colliding objects can bind and become a new compound object moving in a different way and possibly having a different behaviour. In this case we speak of *inelastic* collisions since they are treated with the physical law for that kind of collision.

The Shape Calculus can be used to represent a lot of scenarios at different scales in different fields [5]. However, in this work we use a well-known biological scenario in order to introduce simple running examples that illustrate the semantics of the calculus operators. We consider biochemical reactions occurring inside a cell. Every species of involved molecules has a specific shape and we know from biology that the functions of a molecule are tightly related to its shape. For instance, in enzymatic reactions the functional sites that are active in the enzyme structure, at a given time, determine which substrate (one or two metabolites) can bind the enzyme and proceed to the catalyzed reaction.

Fig. 1 shows a (2D, for simplicity) possible representation of an enzymatic reaction. The larger object represents an enzyme with a shape S_0 and a behaviour B_0 ; S_0 and B_0 together constitute a 3D process $S_0[B_0]$. The 3D process $S_1[B_1]$ represents a metabolite that is close to the given enzyme. Note that portions of the shape surfaces are highlighted: they are the *channels* (an extension of the notion of channels of CCS [17]) that the corresponding 3D processes exhibit to the environment. Each channel has a *name* and an *active surface*. For instance, $\langle a, X \rangle$ is a channel of type a on the active site X of $S_0[B_0]$. The enzyme in Fig. 1 has two channels and its behaviour can be specified as: $B_0 = \langle b, Y \rangle.B'_0 + \langle a, X \rangle.B''_0$. The plus operator represents a non-deterministic choice between two possible communications on the channels. This non-determinism is resolved during the

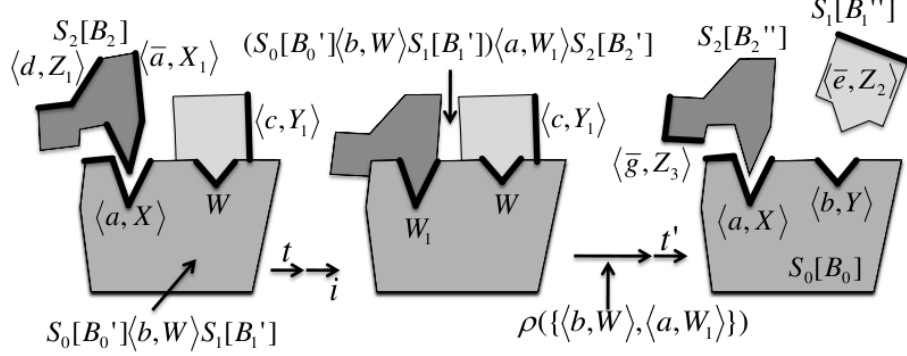


Figure 2: An example of complex formation and subsequent strong-split.

evolution of the system depending on which 3D processes will collide with the enzyme and where.

Following the evolution proposed in the figure, after some time t elapsed (represented by the transition \xrightarrow{t}) and after a detection and resolution of an inelastic collision (transition \rightarrow_i), we get one compound process represented by the term $S_0[B'_0]\langle b, W \rangle S_1[B'_1]$. The bond is established on the surface of contact W and the name b records the type of channels that bound. Note that communication, i.e. *binding*, can only happen if there is a collision between two 3D processes that expose compatible channels (name and co-name à la CCS) on their surface of contact. If the channels were not compatible, the collision would have been treated as elastic and the two 3D processes would have simply bounced. By letting $B'_1 = \langle c, Y_1 \rangle . B''_1$ we allow the component of shape S_1 to open a new channel and to bind with other colling processes.

The third stage of Fig. 1 represents a possible evolution of an enzyme binding with a substrate; it can happen that, for some reason, the bond is loose and the two molecules are free again. To model this kind of behaviour we introduce a special kind of *not urgent* split operation, called *weak-split operation*, that can be delayed of an unspecified time. This is another source of non-determinism in the calculus. Fig. 2 shows what happens when another substrate – that we represent as the 3D process $S_2[B_2]$ with a channel $\langle \bar{a}, X_1 \rangle$ – binds with the compound process $S_0[B'_0]\langle b, W \rangle S_1[B'_1]$ on the common surface W_1 . In the terminology of biochemical reactions, a final complex has been formed. As a consequence, the reaction must proceed and

the products must be released. In our calculus, reactions are represented by *strong-split operations*. Differently from weak-split operations, this kind of split operations cannot be delayed and *must* occur as soon as they are enabled, i.e. when all the involved components can release all the bonds. In this example the involved components are $S_0[B'_0]$, $S_1[B'_1]$ and $S_2[B'_2]$; the set of bonds to be split is $L = \{\langle b, W \rangle, \langle a, W_1 \rangle\}$.

3 3D Shapes

We start by introducing three dimensional shapes as terms of a suitable language, allowing simpler shapes to bind and form more complex shapes. From now on we consider assigned a *global* coordinate system in a three dimensional space represented by \mathbb{R}^3 . Let $\mathbb{P}, \mathbb{V} = \mathbb{R}^3$ be the sets of positions and velocities, respectively, in this coordinate system. Throughout the paper, we assume relative coordinate systems that will always be w.r.t. a certain shape S , i.e. the origin of the relative system is the reference point \mathbf{p} of S . We refer to this relative system as the *local* coordinate system of shape S . If $\mathbf{p} \in \mathbb{P}$ is a position expressed in global coordinates, and $V \subseteq \mathbb{P}$ is a set of points expressed w.r.t. a local coordinate system whose origin is \mathbf{p} , we define $\text{global}(V, \mathbf{p}) = V + \mathbf{p} = \{\mathbf{x} + \mathbf{p} \mid \mathbf{x} \in V\}$ to be the set V w.r.t. the global coordinates. Using local coordinate systems we can express parts of a given shape, such as faces and vertexes, independently from its actual global position.

Definition 1 (Basic Shapes) *A basic shape σ is a tuple $\langle V, m, \mathbf{p}, \mathbf{v} \rangle$ where $V \subseteq \mathbb{P}$ is a convex polyhedron (e.g. a sphere, a cone, a cylinder, etc.)² that represents the set of shape points; $m \in \mathbb{R}^+$, $\mathbf{p} \in \mathbb{P}$ and $\mathbf{v} \in \mathbb{V}$ are, resp., the mass, the centre of mass³ and the velocity of σ . All possible basic shapes are ranged over by σ, σ', \dots . We also define the boundary $\mathcal{B}(\sigma)$ of σ to be the subset of points of V that are on the surface of σ ⁴*

Note that basic shapes are very simple convex shapes. They can be represented by means of suitable and efficient data structures and are handled by the most popular algorithms for motion simulation, collision detection and response [13]. Three dimensional shapes of any form can be approximated

²From a syntactical representation point of view, we assume that V is finitely represented by a suitable data structure, such as a formula or a set of vertices.

³We actually need only a reference point. Thus, any other point in V can be chosen.

⁴Note that we consider only closed shapes, i.e. they contain their boundary.

- with arbitrary precision - by composing basic shapes: the composition of two shapes corresponds to the construction of a third shape by “gluing the two components on a common surface. This concept is generalized by the following definition.

Definition 2 (3D shapes) *The set \mathbb{S} of the 3D shapes, ranged over by S, S', \dots , is generated by the grammar:*

$$S ::= \sigma \mid S \langle X \rangle S$$

where σ is a basic shape and $X \subseteq \mathbb{P}$. If $S = \sigma = \langle V, m, \mathbf{p}, \mathbf{v} \rangle$, we define $\mathcal{P}(S) = V$, $m(S) = m$, $\mathcal{R}(S) = \mathbf{p}$, $\mathbf{v}(S) = \{\mathbf{v}\}$ to be, resp., the set of points, the mass, the reference point and the velocity of S . If $S = S_1 \langle X \rangle S_2$ is a compound shape, then: $\mathcal{P}(S) = \mathcal{P}(S_1) \cup \mathcal{P}(S_2)$, $m(S) = m(S_1) + m(S_2)$, $\mathcal{R}(S) = (m(S_1) \cdot \mathcal{R}(S_1) + m(S_2) \cdot \mathcal{R}(S_2)) / (m(S_1) + m(S_2))$ ⁵ and $\mathbf{v}(S) = \mathbf{v}(S_1) \cup \mathbf{v}(S_2)$. Finally, the boundary of S is defined to be the set $\mathcal{B}(S) = (\mathcal{B}(S_1) \cup \mathcal{B}(S_2)) \setminus \{\mathbf{x} \in \mathbb{P} \mid \mathbf{x} \text{ is interior of } \mathcal{P}(S_1 \langle X \rangle S_2)\}$, where a point $\mathbf{x} \in V \subseteq \mathbb{P}$ is said to be interior if there exists an open ball with centre \mathbf{x} which is completely contained in V .

In this paper we only consider shapes that are *well-formed* according to the following definition.

Definition 3 (Well-formed shapes) *Each basic shape σ is well-formed. A compound shape $S_1 \langle X \rangle S_2$ is well-formed iff:*

1. both S_1 and S_2 are well-formed;
2. the set $X = \mathcal{P}(S_1) \cap \mathcal{P}(S_2)$ is non-empty and equal to $\mathcal{B}(S_1) \cap \mathcal{B}(S_2)$;
3. $\mathbf{v}(S_1 \langle X \rangle S_2)$ is a singleton $\{\mathbf{v}\}$ ⁶ where $\mathbf{v} = \mathbf{v}(S_1) = \mathbf{v}(S_2)$.

Below we say that two shapes S_1 and S_2 *interpenetrate each other* if there exists a point \mathbf{x} that is interior of both $\mathcal{P}(S_1)$ and $\mathcal{P}(S_2)$. In other terms, they *interpenetrate* iff the set $X = \mathcal{P}(S_1) \cap \mathcal{P}(S_2) \neq \emptyset$ is not a subset of $\mathcal{B}(S_1) \cap \mathcal{B}(S_2)$. If $S = S_1 \langle X \rangle S_2$ is well-formed, X is said to be the *surface of contact* between S_1 and S_2 ; moreover, each $\mathbf{x} \in X$ is a *point of contact*.

⁵Again for simplicity, we use the centre of mass as the reference point. Any other point can also be chosen.

⁶With abuse of notation, throughout the paper, we write $\mathbf{v}(S)$ also to refer to the element \mathbf{v} of the singleton $\{\mathbf{v}\}$ as this is not ambiguous when S is well-formed.

Condition (2) guarantees that well-formed compound shapes touch but do not interpenetrate; the surface of contact X is always on the boundary of both S_1 and S_2 . It can be a single point, a segment or a surface, depending on where the two shapes are touching without interpenetrating. Most of the time X is a (subset of) a *feature* of the basic shapes composing the 3D shape, i.e., a face, an edge or a vertex. Condition (3) imposes that all the shapes forming a compound shape have the same velocity; thus, the compound shape moves as a unique body.

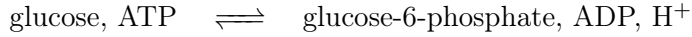
A compound 3D shape S can be represented in a number of different ways by rearranging its basic shapes and surfaces of contact. All these possible representation are ‘equivalent’ w.r.t. the structural congruence defined below.

Definition 4 (Structural Congruence of 3D Shapes) *The structural congruence relation over \mathbb{S} , denoted by \equiv_S , is the smallest relation that satisfies the following rules:*

1. $S_1 \langle X \rangle S_2 \equiv_S S_2 \langle X \rangle S_1$;
2. $(S_1 \langle X \rangle S_2) \langle Y \rangle S_3 \equiv_S S_1 \langle X \rangle (S_2 \langle Y \rangle S_3)$ provided that the surface of contact $Y \subseteq \mathcal{B}(S_2) \cap \mathcal{B}(S_3)$.

The next example shows how the particular features of Shape Calculus can be used to construct a model in a well-known biological scenario.

Example 1 (A Biological Example) *The glycolysis pathway is part of the process by which individual cells produce and consume nutrient molecules. It consists of ten sequential reactions, all catalyzed by a specific enzyme. We focus, in this example, on the first reaction that can be described as*



where an ATP is consumed and a molecule of glucose (GLC) is phosphorylated to glucose 6-phosphate (G6P), releasing an ADP molecule and a positive hydrogen ion (Hydron).

The enzyme catalyzing this first reaction is Hexokinase (HEX). GLC, G6P, ATP, ADP and H^+ are metabolites. Both enzymes and metabolites are autonomous cellular entities that continuously move within the cytoplasm. The transformation of a metabolite into the one that follows in the “pipeline” of the pathway (in this case, GLC into G6P) depends on the meeting (collision in binding sites) of the right enzyme (in this example HEX) with the right metabolites, in this example GLC and ATP. The order of these

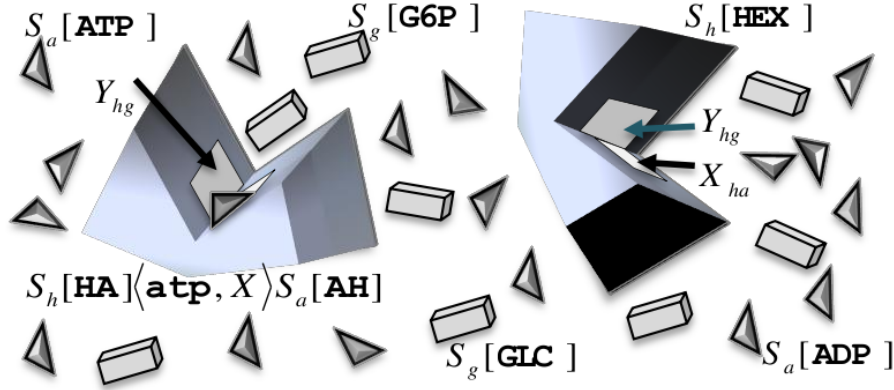


Figure 3: A network of 3D processes for describing the first reaction of the glycolysis pathway.

bindings does not matter. After this binding the reaction takes place and the products⁷ are released in the cytoplasm. A special case occurs when the enzyme has bound one metabolite and an environmental event makes it release the metabolite and not proceed to the completion of the reaction.

We model the shape of HEX, which we denote with S_h , by a polyhedron approximating its real shape and mass, available at public databases (e.g. [1]). Fig. 3 shows a network of 3D processes in which there are two hexokinases and some GLC, G6P, ATP and ADP 3D processes. Note that we use a unique kind of shape for GLC and G6P, denoted by S_g , and another unique kind of shape for ATP and ADP, denoted S_a . They will be distinguished by their behaviours.

Note that modelling of biochemical reactions with our calculus is very different from the usual ODE-based approach. This is because the Shape Calculus embeds concepts and features of a particle-based, individual-based and spatial geometric approach and we wanted to show, in a relatively simple and well-known scenario, how all these concepts and features can be used.

3.1 Trajectories of Shapes

The general idea of the Shape Calculus is to consider a three-dimensional space in which several shapes reside, move and interact. One of the choices

⁷In this example we neglect the hydron.

to be made is how the velocity of each shape changes over time. We believe that a continuous updating of the velocity - that would be a candidate for an ‘as precise as possible approach of modelling - is not a convenient choice. The main reason is the well-known compromise between the benefits of approximation and the complexity of precision. Our choice, quite common also in computer graphics [13], is to approximate a continuous trajectory of a shape with a polygonal chain, i.e. a piecewise linear curve in which each segment is the result of a movement with a constant velocity. The vertices of the chain corresponds to velocity updates.

To this purpose we define a global parameter $\Delta \in \mathbb{R}^+$, called *movement time step*, that represents the maximum period of time after which the velocities of all shapes are updated. The quantification of Δ depends on the desired degree of approximation and also on other parameters connected to collision detection (see Section 3.2). In some situations, the time of updating can be shorter than Δ because, before that time, collisions between moving shapes can occur. These collisions must be resolved and the whole system must re-adapt itself to the new situation. The time domain $\mathbb{T} = \mathbb{R}_0^+$ is then divided into an infinite sequence of time steps t_i such that $t_0 = 0$ and $t_i \leq t_{i-1} + \Delta$ for all $i > 0$. An example in [5] (cf. Section 2) shows in more details how the timeline can be broken up into such time instants.

In the calculus, the updating of velocities is performed by exploiting a function $\text{steer}: \mathbb{T} \rightarrow (\mathbb{S} \hookrightarrow \mathbb{V})$ that describes how the velocity of all existing shapes (i.e. all shapes that are currently moving in the space) at each time t is changed. We assume that, at any given time instant $t \in \mathbb{T}$, $\text{steer } t \ S$ is undefined iff shape S does not exist and, hence, its velocity has not to be changed.

This approach provides us with the maximal flexibility for defining motion. Static shapes are those shapes whose velocity is always zero⁸. A gravity field can be simulated by updating the velocities according to the gravity acceleration. A Brownian motion can be simulated by choosing a random 3D direction and then defining the length of the vector w.r.t. the mass and/or the volume of the shape. In this paper, we do not consider movements due to rotations. However, this kind of movements can be easily added to our shapes by assigning an angular velocity and a moment of inertia to a shape and then by performing a compound motion of translation and rotation along the movement time step.

⁸To represent walls, we also need to assign an infinite value to the mass of these objects, otherwise they can be moved anyway due to collisions.

Let us define now some useful notation and properties.

Definition 5 (Evolution of shapes over time) *Let $S \in \mathbb{S}$ and $t \in \mathbb{T}$; $S + t$, i.e. the shape S after t time units, is defined by induction on S :*

$$\begin{aligned} \text{Basic:} \quad & \langle V, m, \mathbf{p}, \mathbf{v} \rangle + t = \langle V + (t \cdot \mathbf{v}), m, \mathbf{p} + (t \cdot \mathbf{v}), \mathbf{v} \rangle \\ \text{Comp:} \quad & (S_1 \langle X \rangle S_2) + t = (S_1 + t) \langle X + t \cdot \mathbf{v}(S) \rangle (S_2 + t) \end{aligned}$$

Definition 6 (Updating shape velocity) *Let $S \in \mathbb{S}$ and $\mathbf{w} \in \mathbb{V}$. We define the shape $S[\![\mathbf{w}]\!]$, i.e. S whose velocity is updated with \mathbf{w} , as follows:*

$$\begin{aligned} \text{Basic:} \quad & \langle V, m, \mathbf{p}, \mathbf{v} \rangle[\![\mathbf{w}]\!] = \langle V, m, \mathbf{p}, \mathbf{w} \rangle \\ \text{Comp:} \quad & (S_1 \langle X \rangle S_2)[\![\mathbf{w}]\!] = (S_1[\![\mathbf{w}]\!]) \langle X \rangle (S_2[\![\mathbf{w}]\!]) \end{aligned}$$

The following result comes directly from Def. 3.

Proposition 1 *Let $S \in \mathbb{S}$, $t \in \mathbb{T}$ and $\mathbf{w} \in \mathbb{V}$. If S is well-formed then $S + t$ and $S[\![\mathbf{w}]\!]$ are well-formed.*

Our intent is to represent a lot of shapes moving simultaneously in space as described above. Inevitably, this scenario produces collisions between shapes when their trajectories encounter.

3.2 Collision Detection

There is a rich literature on collision detection systems, as this problem is fundamental in popular applications like computer games. Good introductions to existing methods for efficient collision detection are available and we refer to Ericson [13] and references therein for a detailed treatment.

For our purposes, it is sufficient to define an interface between our calculus and a typical collision detection system. We can then choose one of them according to their different characteristics, e.g. their applicability in large-scale environments or their robustness. It must be said, however, that the choice of the collision detection system may influence the kind of basic (or compound) shapes we can use, as, for instance, some systems may require the use of only convex shapes to be more efficient⁹.

Typically, a collision detection algorithm assumes to start in a situation in which shapes do not interpenetrate. Then it tries to move all the shapes

⁹The basic shapes that we consider in Definition 1 are typically accepted by most of the collision detection systems.

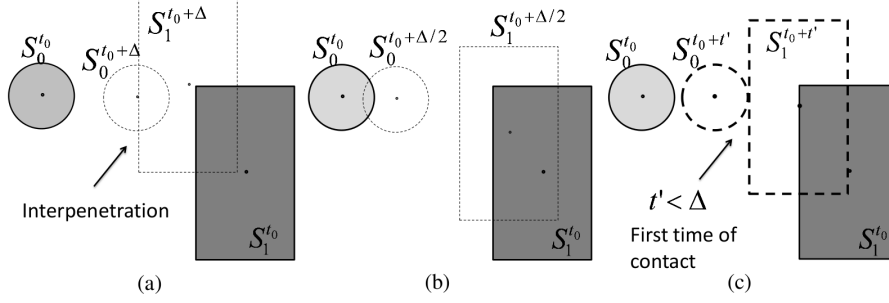


Figure 4: Some steps to determine the first time of contact between two shapes.

of a little¹⁰ time step - that we have already introduced as the movement time step Δ - and check if interpenetrations occurred¹¹. If so, it tries to consider only half of the original time step and repeat the interpenetration check, i.e. it performs a binary search of the *first time of contact* t between two or more shapes, with some degree of approximation. Fig. 4 shows these steps. In case (a) the passage of the whole Δ results in an interpenetration. Then, in (b) the passage of $\Delta/2$ is tried resulting into a non-contact. After some iterations the situation in (c) is reached.

In addition to the first time of contact, a collision detection algorithm usually outputs the shapes that are colliding, i.e. are touching without interpenetrating after t , and some information about the surfaces or points of contact. We now define precisely what we expect to obtain from a collision detection system.

Definition 7 (First time of contact) *Let I be a non-empty finite set of indexes and let $\{S_i\}_{i \in I}$ be a set of well-formed shapes such that for all $i, j \in I$, S_i and S_j do not interpenetrate (see Def. 3). The first time of contact of the shapes S_i , denoted $\text{Ftoc}(\{S_i\}_{i \in I})$, is a number $t \in \mathbb{T}$ such that:*

¹⁰The time step must be chosen little enough to avoid that any two shapes, at their possible maximum velocity, start in a non-interpenetration state, engage in an interpenetration and exit from the interpenetration state in a single time step duration.

¹¹Typically, the major efforts of optimization are concentrated in this step since the number of checks is, in the worst case, $O(N^2)$ - where N is the number of shapes in the space - but the shapes that are likely to collide are only those that are very close to each other.

1. for all $t' \in \mathbb{T}$, $0 \leq t' \leq t$ and for all $i, j \in I$, $S_i + t'$ and $S_j + t'$ do not interpenetrate;
2. there exist $i, j \in I$, with $i \neq j$, such that $\mathcal{B}(S_i + t) \cap \mathcal{B}(S_j + t) \neq \emptyset$, i.e., some shapes are touching at t ;
3. for all $\epsilon > 0$ there exists δ , $0 < \delta < \epsilon$, and $i, j \in I$, with $i \neq j$, such that $S_i + (t + \delta)$ and $S_j + (t + \delta)$ interpenetrate, i.e., in t some shapes are touching and any further movement makes them to interpenetrate.

Note that such a definition allows shapes that are touching without interpenetrating, and with velocities that do not make them to interpenetrate (e.g., the same velocity), to move without triggering a first time of contact. This will be useful in Section 5 when we split previously compound shapes. Indeed, after the split these shapes have the same velocity and, hence, do not affect the next first time of contact.

Definition 8 (Collision information) Let $\{S_i\}_{i \in I}$ be a set of well-formed shapes and let $t = \text{Ftoc}(\{S_i\}_{i \in I})$ be their first time of contact. The set of colliding shapes after time t is denoted by $\text{colliding}(\{S_i\}_{i \in I}) \subseteq \mathbb{S} \times \mathbb{S} \times \wp(\mathbb{P})$. A tuple $\langle S_i, S_j, X \rangle \in \text{colliding}(\{S_i\}_{i \in I})$ iff:

1. $\mathcal{P}(S_i + t) \cap \mathcal{P}(S_j + t)$ is non-empty and is equal to $\mathcal{B}(S_i + t) \cap \mathcal{B}(S_j + t)$;
2. for all $\epsilon > 0$ there exists δ , $0 < \delta < \epsilon$, such that $S_i + (t + \delta)$ and $S_j + (t + \delta)$ interpenetrate.

3.3 Collision Response

In this section, we briefly discuss the problem of *collisions response* [15], i.e. how collisions, once detected, can be resolved. We distinguish between *elastic* collisions (those in which there is no loss in kinetic energy) and *perfectly inelastic* ones (in which kinetic energy is fully dissipated)¹². After an elastic collision, the two shapes will proceed independently to each other but their velocities will be changed according to the laws for conservation of linear momentum and kinetic energy - Equations (1)-(2) in Def. 9. On the contrary, two shapes that collide inelastically will bind together and will move as a *unique* body whose velocity is given by the law for conservation of linear momentum only - Equation (3), in Def. 9.

¹²Other different kinds of collisions can be easily added to the calculus provided that the corresponding collision response laws are given.

Definition 9 (Collisions) Let $S_1, S_2 \in \mathbb{S}$ and let $X \subseteq \mathbb{P}$ be a surface of contact between them. If X is neither an edge nor a vertex of S_1 , the velocities \mathbf{w}_1 and \mathbf{w}_2 of these shapes after an elastic collision in X are given by:

$$(1) \quad \mathbf{w}_1 = \mathbf{v}(S_1) - \frac{\lambda}{m(S_1)} \cdot \mathbf{n} \quad (2) \quad \mathbf{w}_2 = \mathbf{v}(S_2) + \frac{\lambda}{m(S_2)} \cdot \mathbf{n}$$

where \mathbf{n} is the normal of contact away from $X \subseteq \mathcal{B}(S_1)$, i.e. the unit vector perpendicular to the face of S_1 that contains X , and

$$\lambda = 2 \frac{m(S_1) m(S_2)}{m(S_1) + m(S_2)} \frac{\mathbf{v}(S_1) \cdot \mathbf{n} - \mathbf{v}(S_2) \cdot \mathbf{n}}{\mathbf{n} \cdot \mathbf{n}}$$

If X is either an edge or a vertex of S_1 , \mathbf{n} is the normal of contact away from the shape S_2 and velocities \mathbf{w}_1 and \mathbf{w}_2 are obtained by means of symmetric equations. In both cases, we write $S_1 \xleftrightarrow{X}_e S_2$ to denote the pair of velocities $(\mathbf{w}_1, \mathbf{w}_2)$. If S_1 and S_2 collide inelastically in the surface of contact X , they will bind together as a unique body whose velocity (denoted with $S_1 \xleftrightarrow{X}_i S_2$) is given by:

$$(3) \quad \mathbf{v} = \frac{m(S_1)}{m(S_1) + m(S_2)} \cdot \mathbf{v}(S_1) + \frac{m(S_2)}{m(S_1) + m(S_2)} \cdot \mathbf{v}(S_2)$$

4 3D Processes

In this section we introduce the timed process algebra whose terms describe the *internal behaviour* of our 3D shapes. This is a variation of Timed CCS [21], where basic actions provide information about binding capability and splits of shape bonds. We use the following notation. Let $\Lambda = \{a, b, \dots\}$ be a countably infinite set of *channels names* (names, for short) and $\bar{\Lambda} = \{\bar{a} \mid a \in \Lambda\}$ its complementation; by convention we assume $\bar{\bar{a}} = a$ for each name a . Elements in $\mathcal{A} = \Lambda \cup \bar{\Lambda}$ are ranged over by α, β, \dots .

Binding capabilities are represented by *channels*, i.e. pairs $\langle \alpha, X \rangle$ where $\alpha \in \mathcal{A}$ is a name and X is a *surface of contact*. Intuitively, a surface of contact is a portion of space (usually, a subset of the boundary of a given 3D shape) where the channel itself is active and where binding with other processes are possible. Names introduce a notion of compatibility between channels useful to distinguish between elastic and inelastic collisions. If $\beta = \bar{\alpha}$ and $X \cap Y \neq \emptyset$ we say that the channels $\langle \alpha, X \rangle$ and $\langle \beta, Y \rangle$ are *compatible*,

written $\langle \alpha, X \rangle \sim \langle \beta, Y \rangle$. Otherwise, $\langle \alpha, X \rangle$ and $\langle \beta, Y \rangle$ are *incompatible* which we write as $\langle \alpha, X \rangle \not\sim \langle \beta, Y \rangle$.

We also introduce two different kinds of actions that represent splits of shape bonds. In particular, we distinguish between *weak-split actions* of the form $\omega(\alpha, X)$ and *strong-split actions* of the form $\rho(\alpha, X)$. With an abuse of notation, two weak-split actions $\omega(\alpha, X)$ and $\omega(\beta, Y)$ (as similarly for the strong-split actions $\rho(\alpha, X)$ and $\rho(\beta, Y)$) are compatible if so are the channels $\langle \alpha, X \rangle$ and $\langle \beta, Y \rangle$. We will see that a synchronization between a pair of compatible weak-split actions results in a weak-split operation, while synchronizations between multiple pairs of compatible strong-split actions correspond to a strong-split operation. These operations behave differently w.r.t. to *time passing* since the latter operation cannot time pass further, while the former one can be arbitrarily delayed.

Let \mathcal{C} be the set of all channels, $\omega(\mathcal{C}) = \{\omega(\alpha, X) \mid \langle \alpha, X \rangle \in \mathcal{C}\}$ and $\rho(\mathcal{C}) = \{\rho(\alpha, X) \mid \langle \alpha, X \rangle \in \mathcal{C}\}$ be the sets of *weak-split actions* and *strong-split actions*, resp. Our processes perform basic and *atomic* actions that belong to the set $\mathbf{Act} = \mathcal{C} \cup \omega(\mathcal{C}) \cup \rho(\mathcal{C})$ whose elements are ranged over by μ, μ', \dots . We finally assume a countably infinite collection \mathcal{K} of *process name* or *process constants*.

Definition 10 (Shape behaviours)

The set of shape behaviours, denoted by \mathbb{B} , is generated by the following grammar:

$$B ::= \text{nil} \mid \langle \alpha, X \rangle.B \mid \omega(\alpha, X).B \mid \rho(L).B \mid \epsilon(t).B \mid B + B \mid K$$

where $\langle \alpha, X \rangle \in \mathcal{C}$, $L \subseteq \mathcal{C}$ (non-empty) whose elements are pairwise incompatible (i.e. for each pair $\langle \alpha, X \rangle, \langle \beta, Y \rangle \in L$ it is $\langle \alpha, X \rangle \not\sim \langle \beta, Y \rangle$), $t \in \mathbb{T}$ and K is a process name in \mathcal{K} .

A brief description of our operators now follows. nil is the Nil-behaviour, it can not perform any action but can let time pass without limits. A trailing nil will often be omitted, so e.g. we write $\langle a, X \rangle.\omega(a, X)$ to abbreviate $\langle a, X \rangle.\omega(a, X).\text{nil}$. $\langle \alpha, X \rangle.B$ and $\omega(\alpha, X).B$ are (action-)prefixing known from CCS; they evolve in B by performing the actions $\langle \alpha, X \rangle$ and $\omega(\alpha, X)$, resp. $\langle \alpha, X \rangle.B$ exhibits a binding capability along the channel $\langle \alpha, X \rangle$, while $\omega(\alpha, X).B$ models the behaviour of a shape that, before evolving in B , wants to split a *single* bond established via the channel $\langle \alpha, X \rangle$. $\rho(L).B$ is the *strong-split operator*; it can evolve in B only after the execution of *all* strong-split actions $\rho(\alpha, X)$ with $\langle \alpha, X \rangle \in L$. The delay-prefixing operator $\epsilon(t).B$

$\text{NIL}_t \frac{}{\text{nil} \xrightarrow{t} \text{nil}}$	$\text{PREF}_t \frac{\mu \in \mathcal{C} \cup \omega(\mathcal{C})}{\mu.B \xrightarrow{t} \mu.B}$	$\text{STR}_t \frac{}{\rho(L).B \xrightarrow{t} \rho(L).B}$
$\text{SUM}_t \frac{B_1 \xrightarrow{t} B'_1 \quad B_2 \xrightarrow{t} B'_2}{B_1 + B_2 \xrightarrow{t} B'_1 + B'_2}$	$\text{DEL}_t \frac{t' \geq t}{\epsilon(t').B \xrightarrow{t} \epsilon(t' - t).B}$	
	$\text{DEF}_t \frac{B \xrightarrow{t} B'}{K \xrightarrow{t} B'} \quad \text{if } K \stackrel{\text{def}}{=} B$	

Table 1: Temporal behaviour of \mathbb{B} 's terms

(see [21]) introduces time delays in 3D processes; $t \in \mathbb{T}$ is the amount of time that has to elapse before the idling time is over (see rules DELAY_t and DELAY_a in Tables 1 and 2). Finally, $B_1 + B_2$ models a non-deterministic choice between B_1 and B_2 and K is a process definition.

In the remainder of this paper, we use processes in \mathbb{B} to define the internal behaviour of 3D shapes. For this reason, we assume that sites in binding capabilities, as well as in weak- and strong-split actions, are expressed w.r.t. the *local* coordinate system whose origin is the reference point of the shape where they are embedded in.

Definition 11 (Operational semantics of shape behaviours) *The SOS-rules that define the temporal transition relations $\xrightarrow{t} \subseteq (\mathbb{B} \times \mathbb{B})$ for $t \in \mathbb{T}$, that describe how shape behaviours evolve by letting time t pass, are provided in Table 1. We write $B \xrightarrow{t} B'$ if $(B, B') \in \xrightarrow{t}$ and $B \xrightarrow{t}$ if there is $B' \in \mathbb{B}$ such that $(B, B') \in \xrightarrow{t}$. Similar conventions will apply later on. Rules in Table 2 define the action transition relations $\xrightarrow{\mu} \subseteq (\mathbb{B} \times \mathbb{B})$ for $\mu \in \text{Act}$. These transitions describe which basic actions a shape behaviour can perform.*

Most of the rules in Table 1 are those provided in [21]. Rules PREF_t and STR_t state that processes like $\langle \alpha, X \rangle.B$, $\omega(\alpha, X).B$ and $\rho(L).B$ can be arbitrarily delayed. The only rules in Table 2 worth noting are those defining the functional behaviour of the strong-split operator. By Rules STR_1 and STR_2 , if $\langle \alpha, X \rangle \in L$ then $\rho(L).B$ can do a $\rho(\alpha, X)$ -action and evolve either in B (if $L = \{\langle \alpha, X \rangle\}$) or in $\rho(L \setminus \{\langle \alpha, X \rangle\}).B$ (otherwise). Rule STR_3

$\text{PREF}_a \frac{\mu \in \mathcal{C} \cup \omega(\mathcal{C})}{\mu.B \xrightarrow{\mu} B}$	$\text{DEL}_a \frac{B \xrightarrow{\mu} B'}{\epsilon(0).B \xrightarrow{\mu} B'}$	$\text{SUM}_a \frac{B_1 \xrightarrow{\mu} B'}{B_1 + B_2 \xrightarrow{\mu} B'}$
$\text{DEF}_a \frac{B \xrightarrow{\mu} B'}{K \xrightarrow{\mu} B'}$	if $K \stackrel{\text{def}}{=} B$	$\text{STR}_1 \frac{L = \{\langle \alpha, X \rangle\}}{\rho(L).B \xrightarrow{\rho(\alpha, X)} B}$
$\text{STR}_2 \frac{L = \{\langle \alpha, X \rangle\} \cup L' \quad L' \neq \emptyset}{\rho(L).B \xrightarrow{\rho(\alpha, X)} \rho(L').B}$		$\text{STR}_3 \frac{B \xrightarrow{\rho(\alpha, X)} B'}{\rho(L).B \xrightarrow{\rho(\alpha, X)} \rho(L).B'}$

Table 2: Functional behaviour of \mathbb{B} -terms

is needed to handle arbitrarily nested terms, e.g. $\rho(\{\langle a, X \rangle\}).\rho(\{\langle \bar{b}, Y \rangle\}).B$. Other rules are as expected.

Now we are ready to define 3D processes, i.e. simple or compound shapes whose behaviour is expressed by a process in \mathbb{B} .

Definition 12 (3D processes) *The set 3DP of 3D processes is generated by the following grammar:*

$$P ::= S[B] \mid P \langle a, X \rangle P$$

where $S \in \mathbb{S}$, $B \in \mathbb{B}$, $a \in \Lambda$ and X is a non-empty subset of \mathbb{P} . The shape of each $P \in 3\text{DP}$ is defined by induction on P as follows:

Basic: $\text{shape}(S[B]) = S$

Comp: $\text{shape}(P \langle a, X \rangle Q) = \text{shape}(P) \langle X \rangle \text{shape}(Q)$

We also define $\mathbf{v}(P) = \mathbf{v}(\text{shape}(P))$ and $\mathcal{B}(P) = \mathcal{B}(\text{shape}(P))$. Below we often write $P \xleftarrow{X} \rightarrow_i Q$ and $P \xleftarrow{X} \rightarrow_e Q$ as shorthand for $\text{shape}(P) \xleftarrow{X} \rightarrow_i \text{shape}(Q)$ and $\text{shape}(P) \xleftarrow{X} \rightarrow_e \text{shape}(Q)$, resp. Finally, $P[\mathbf{v}]$ is the 3D process we obtain by updating P 's velocity as follows:

Basic: $(S[B])[\mathbf{v}] = (S[\mathbf{v}])[B]$

Comp: $(P \langle a, X \rangle Q)[\mathbf{v}] = (P[\mathbf{v}]) \langle a, X \rangle (Q[\mathbf{v}])$

We finally write $\text{steer } tP$ to denote $P[\text{steer } t \text{shape}(P)]$. We say that a basic process $S[B]$ is well-formed iff the shape S is well-formed and, for each

$X \subseteq \mathbb{P}$ that occurs in B , $\text{global}(X, \mathcal{R}(S)) \subseteq \mathcal{B}(S)$. A compound process $P \langle a, X \rangle Q$ is well-formed iff P and Q are well-formed, $\mathbf{v}(P) = \mathbf{v}(Q)$ and the site X expressed w.r.t a global coordinate system is a non-empty subset of $\mathcal{B}(P) \cap \mathcal{B}(Q)$. Note that this also means that the set $\mathcal{P}(P) \cap \mathcal{P}(Q)$ is non-empty and equal to $\mathcal{B}(P) \cap \mathcal{B}(Q)$. Later on in this paper we only consider well-formed processes.

We can state the following proposition as an easy consequence of shapes and 3D processes well-formedness.

Proposition 2 *For each $P \in \text{3DP}$ well-formed, $\text{shape}(P)$ is well-formed.*

Let us model the molecules involved in the reaction of Example 1 as 3D processes.

Example 2 (3D Processes for HEX, GLC and ATP) *An Hexokinase molecule is modeled as $\text{HEX} \stackrel{\text{def}}{=} S_h[\text{HEX}]$ where:*

$\text{HEX} = \langle \text{atp}, X_{ha} \rangle. \text{HA} + \langle \text{glc}, X_{hg} \rangle. \text{HG}$,
 $\text{HA} = \omega(\text{atp}, X_{ha}). \text{HEX} + \epsilon(t_h). \langle \text{glc}, X_{hg} \rangle. \rho(\{\langle \text{atp}, X_{ha} \rangle, \langle \text{glc}, Y_{hg} \rangle\}). \text{HEX}$,
 $\text{HG} = \omega(\text{glc}, X_{hg}). \text{HEX} + \epsilon(t_h). \langle \text{atp}, X_{ha} \rangle. \rho(\{\langle \text{atp}, X_{ha} \rangle, \langle \text{glc}, Y_{hg} \rangle\}). \text{HEX}$,
and X_{ha}, Y_{hg} are the surfaces of contact shown in Fig. 3. $\text{ATP} = S_a[\text{ATP}]$ models an ATP molecule where:

$$\text{ATP} = \langle \overline{\text{atp}}, X_{ah} \rangle. (\epsilon(t_a). \rho(\{\langle \overline{\text{atp}}, X_{ah} \rangle\}). \text{ADP} + \omega(\overline{\text{atp}}, X_{ah}). \text{ATP})$$

and the surface of contact X_{ah} is the whole boundary $\mathcal{B}(S_a)$. The process modelling a molecule of glucose is similar: $\text{GLC} = S_g[\text{GLC}]$ where

$$\text{GLC} = \langle \overline{\text{glc}}, X_{gh} \rangle. (\epsilon(t_g). \rho(\{\langle \overline{\text{glc}}, X_{gh} \rangle\}). \text{G6P} + \omega(\overline{\text{glc}}, X_{gh}). \text{GLC})$$

We leave unspecified the behaviours G6P and ADP.

HEX has two channels $\langle \text{atp}, X_{ha} \rangle$ and $\langle \text{glc}, Y_{hg} \rangle$ to bind, resp., with an ATP and a GLC molecule. By performing an action $\langle \text{atp}, X_{ha} \rangle$, HEX evolves in HA. HA can perform either a weak-split action $\omega(\text{atp}, X_{ha})$ to come back to HEX, or can wait at most t_h units of time, perform $\langle \text{glc}, Y_{hg} \rangle$ and then evolve in $\rho(\{\langle \text{atp}, X_{ha} \rangle, \langle \text{glc}, Y_{hg} \rangle\}). \text{HEX}$. Now, two strong-split actions are enabled after which we come back to HEX. Notice that, after an action $\langle \text{glc}, Y_{hg} \rangle$, HEX becomes HG that behaves symmetrically.

An ATP molecule performs a $\langle \overline{\text{atp}}, X_{ah} \rangle$ -action, waits t_r units of time, and then can release the bond established on the channel $\langle \overline{\text{atp}}, X_{ah} \rangle$ – and thus return free as ATP – or can participate in the reaction and become an ADP. As we will see in Section 5, the result is the split of the complex in the

three original shapes whose behaviours are HEX, ADP and G6P, resp. We omit the description of the behaviour of GLC since it is similar to that of ATP.

We are ready to define the timed operational semantics of 3D processes.

Definition 13 (Transitional semantics of 3D processes) *Rules in Table 3 define the transition relations $\xrightarrow{t} \subseteq (3DP \times 3DP)$ for $t \in \mathbb{T}$, and $\xrightarrow{\mu} \subseteq 3DP \times 3DP$ for $\mu \in \text{Act}$. Two 3D processes P and Q are said to be compatible, written $P \sim Q$, if $P \xrightarrow{\langle \alpha, X \rangle}$ and $Q \xrightarrow{\langle \bar{\alpha}, Y \rangle}$ for some compatible channels $\langle \alpha, X \rangle$ and $\langle \bar{\alpha}, Y \rangle$; otherwise, P and Q are incompatible that we denote with $P \not\sim Q$. Below, we often write $P \xrightarrow{\rho}$ and $P \xrightarrow{\omega}$ as shorthand for $P \xrightarrow{\rho(\alpha, X)}$ and $P \xrightarrow{\omega(\alpha, X)}$, resp., for any $\langle \alpha, X \rangle$.*

Essentially, rules in Table 3 say that a 3D process inherits its functional and temporal behaviour from the \mathbb{B} -terms defining its internal behaviour. But now sites of binding capabilities and split actions are expressed w.r.t. a *global* coordinate system (see rules BASIC_c and BASIC_s). For simplicity, we have omitted a rule defining which weak-split action a basic process can perform. This can be obtained from rule BASIC_w by replacing each $\rho()$ -action with a corresponding $\omega()$ -action. It is worth noting that, due to rule COMP_{a2} , some of the $\langle \alpha, Y \rangle$ -actions performed by P (by Q) can be prevented in $P \langle a, X \rangle Q$ since, due to binding a part (or all) of Y has become interior because it is covered by a piece of Q (of P respectively) the surface of contact $Y \not\subseteq \mathcal{B}(P \langle a, X \rangle Q)$ and, hence the corresponding channel is no more active. We have also omitted symmetric rules for COMP_{a1} and COMP_{a2} for the actions of Q .

The following proposition (see Appendix B for the proof) shows that 3D processes well-formedness is closed w.r.t. transitions \xrightarrow{t} and $\xrightarrow{\mu}$.

Proposition 3 *Let $P \in 3DP$ well-formed. Either $P \xrightarrow{t} Q$ or $P \xrightarrow{\mu} Q$ implies $Q \in 3DP$ well-formed.*

At this stage a key observation is that the operational rules in Table 3 do not allow synchronization between components of compound process that proceed independently to each other. Consider, as an example,

$$P = S_p[\rho(\{a, X_p\}).B_p],$$

	$\text{BASIC}_t \frac{B \overset{t}{\rightsquigarrow} B'}{S[B] \overset{t}{\rightsquigarrow} (S+t)[B']}$
COMP_t	$\frac{P \overset{t}{\rightsquigarrow} P' \quad Q \overset{t}{\rightsquigarrow} Q' \quad X' = X + (t \cdot \mathbf{v}(P))}{P \langle a, X \rangle Q \overset{t}{\rightsquigarrow} P' \langle a, X' \rangle Q'}$
BASIC_c	$\frac{B \xrightarrow{\langle \alpha, X \rangle} B' \quad Y = \text{global}(X, \mathcal{R}(S))}{S[B] \xrightarrow{\langle \alpha, Y \rangle} S[B']}$
BASIC_s	$\frac{B \xrightarrow{\rho(\alpha, X)} B' \quad Y = \text{global}(X, \mathcal{R}(S))}{S[B] \xrightarrow{\rho(\alpha, Y)} S[B']}$
COMP_s	$\frac{P \xrightarrow{\rho(\alpha, Y)} P'}{P \langle a, X \rangle Q \xrightarrow{\rho(\alpha, Y)} P' \langle a, X \rangle Q}$
COMP_w	$\frac{P \xrightarrow{\omega(\alpha, Y)} P'}{P \langle a, X \rangle Q \xrightarrow{\omega(\alpha, Y)} P' \langle a, X \rangle Q}$
COMP_c	$\frac{P \xrightarrow{\langle \alpha, Y \rangle} P' \quad Y \subseteq \mathcal{B}(P \langle a, X \rangle Q)}{P \langle a, X \rangle Q \xrightarrow{\langle \alpha, Y \rangle} P' \langle a, X \rangle Q}$

Table 3: Functional and temporal behaviour of 3DP-terms

$$Q = S_q[\rho(\{\bar{a}, X_q\}).B_q], \text{ and } P \langle a, X \rangle Q$$

where $X = X'_p \cap X'_q$ and, for each $i \in \{p, q\}$, X'_i is the site X_i w.r.t. a global coordinate system, i.e. $X'_i = \text{global}(X_i, \mathcal{R}(S_i))$. As stand-alone processes, P and Q can perform two compatible strong-split actions, namely $\rho(a, X'_p)$ and $\rho(\bar{a}, X'_q)$ and evolve, resp., in $S_p[B_p]$ and $S_q[B_q]$. As a consequence, $P \langle a, X \rangle Q$ becomes either $S_p[B_p] \langle a, X \rangle Q$ or $P \langle a, X \rangle S_q[B_q]$.

But these actions are compatible and, according to the intuition given so far, P and Q have to synchronize on their execution in order to split the bond $\langle a, X \rangle$. In other terms, a strong-split operation is enabled; such an operation must be performed before time can pass further and must produce as a result two independent 3D processes, i.e. the *network* of 3D processes (see Section 5) that contains both $S_p[B_p]$ and $S_q[B_q]$. Similarly, we would allow synchronizations between compatible weak-split action in order to perform a weak-split operation. To properly deal with this kind of behaviours some technical details are still needed. We first allow synchronization on compatible split actions by introducing the transition relations $\xRightarrow{\rho(a, X)}$ and $\xRightarrow{\omega(a, X)}$. Intuitively, we want that $P \langle a, X \rangle Q \xRightarrow{\rho(a, X)} S_p[B_p] \langle a, X \rangle S_q[B_q]$. Now, we can ‘physically’ remove the bond $\langle a, X \rangle$ (this will be done by exploiting the function `split` over 3D processes we provide in the next section) and obtain the network of processes we are interested in.

Definition 14 (Semantics of strong and weak splits) *The SOS-rules that define the transition relations $\xRightarrow{\rho(a, X)} \subseteq \text{3DP} \times \text{3DP}$ where $\rho(a, X) \in \rho(\mathcal{C})$ are given in Table 4. As usual, symmetric rules have been omitted. We also omit the rules defining the transition relations $\xRightarrow{\omega(a, X)} \subseteq \text{3DP} \times \text{3DP}$ for $\omega(a, X) \in \omega(\mathcal{C})$ since these can be obtained from those in Table 4 by replacing each action $\rho(-)$ with the corresponding action $\omega(-)$.*

Recall that strong-split operations require simultaneous split of multiple bonds. In this case, all the components involved in the reaction must *all together* be ready to synchronize on a proper set of compatible strong-split actions. Consider a more complex example

$$P = S_p[\rho(\{\langle a, X_p \rangle, \langle \bar{b}, Y_p \rangle\}).B_p],$$

$$Q = S_q[\rho(\{\langle \bar{a}, X_q \rangle\}).B_q],$$

$$R = S_r[\rho(\{\langle b, Y_r \rangle\}).B_r], \text{ and } (P \langle a, X \rangle Q) \langle b, Y \rangle R$$

where $X = X'_p \cap X'_q$ and $Y = Y'_p \cap Y'_r$ (also in this case we write X'_i , for $i \in \{p, q, r\}$, to represent the site X_i in a global coordinate system). P can synchronize with Q and R to split, resp., the bonds $\langle a, X \rangle$ and $\langle b, Y \rangle$. Indeed, rules in Table 4 implies that

$$\begin{aligned} (P \langle a, X \rangle Q) \langle b, Y \rangle R &\xRightarrow{\rho(a, X)} (S_p[\rho(\{\langle \bar{b}, Y_p \rangle\}).B_p] \langle a, X \rangle S_q[B_q]) \langle b, Y \rangle R \\ &\xRightarrow{\rho(b, Y)} (S_p[B_p] \langle a, X \rangle S_q[B_q]) \langle b, Y \rangle S_r[B_r] \end{aligned}$$

After that, all ‘pending strong-split requests’ of $(P \langle a, X \rangle Q) \langle b, Y \rangle R$ are satisfied. We say that such a compound process is *able to complete a reaction*. If it was, for instance, $R = S_r[\epsilon(t). \rho(\{\langle \bar{b}, Y_r \rangle\}).B_r]$ then $(P \langle a, X \rangle Q) \langle b, Y \rangle R$ would *not* have been able to complete a reaction, since (at least) one involved component, i.e. R , would not have been able to contribute to the reaction before than t units of time. In such a case, the bonds can not be spit at all. This concept is formalized by the definition below.

Definition 15 (Bonds of 3DP-terms) *The function $\text{bonds} : 3\text{DP} \rightarrow \wp(\mathcal{C})$ returns the set of bonds that are currently established in P . It can be defined by induction on $P \in 3\text{DP}$:*

$$\begin{aligned} \text{Basic:} \quad & \text{bonds}(S[B]) = \emptyset \\ \text{Comp:} \quad & \text{bonds}(P \langle a, X \rangle Q) = \text{bonds}(P) \cup \text{bonds}(Q) \cup \{\langle a, X \rangle\} \end{aligned}$$

By an easy induction on P we can prove that $P \xRightarrow{\rho(a, X)}$ implies $\langle a, X \rangle \in \text{bonds}(P)$. Moreover, we say that $P \in 3\text{DP}$ is able to complete a reaction, which we write as $P \searrow$, iff either (1) $P \not\rightarrow$, or (2) $P \xRightarrow{\rho(a, X)} Q$ for some $\rho(a, X)$ and Q such that $Q \searrow$.

$\begin{array}{c} \text{STRSYNC} \frac{P \xrightarrow{\rho(\alpha, X_p)} P' \quad Q \xrightarrow{\rho(\bar{\alpha}, X_q)} Q' \quad \alpha \in \{a, \bar{a}\} \quad X = X_p \cap X_q}{P \langle a, X \rangle Q \xRightarrow{\rho(a, X)} P' \langle a, X \rangle Q'} \\ \\ \text{STRPAR} \frac{P \xRightarrow{\rho(b, Y)} P'}{P \langle a, X \rangle Q \xRightarrow{\rho(b, Y)} P' \langle a, X \rangle Q} \end{array}$

Table 4: Transitional semantics for strong-split actions

Finally, if P is able to complete a reaction and there is at least a bond that has to be strongly split (i.e. if $P \not\rightarrow$ does not hold), a reaction can actually take place and, as a consequence, time cannot pass further. Below we restrict the timed operational semantics of 3D processes as it has been defined in Def. 13 in order to take this aspect into account.

Definition 16 *Let $P \in \mathbf{3DP}$. We say that $P \xrightarrow{t} Q$ if $P \rightsquigarrow^t Q$ and either $P \not\rightarrow$ or P is not able to complete a reaction.*

Proposition 4 states that the function **split** is well-defined up to structural congruence over 3D processes we define below.

Definition 17 (Structural congruence over 3D processes) *We define the structural congruence over processes in $\mathbf{3DP}$, which we denote by \equiv_P , as the smallest relation that satisfies the following axioms:*

- $S[B] \equiv_P S'[B]$ provided that $S \equiv_S S'$;
- $P \langle a, X \rangle Q \equiv_P Q \langle a, X \rangle P$;
- $P \equiv_P Q$ implies $P \langle a, X \rangle R \equiv_P Q \langle a, X \rangle R$;
- $Y \subseteq \mathcal{B}(Q) \cap \mathcal{B}(R)$ implies $(P \langle a, X \rangle Q) \langle b, Y \rangle R \equiv_P P \langle a, X \rangle (Q \langle b, Y \rangle R)$.

Proposition 4 *Let $P \in \mathbf{3DP}$ well-formed. If $\langle a, X \rangle \in \text{bonds}(P)$ there is a well-formed 3D process $Q \langle a, X \rangle R \equiv_P P$.*

We also need the following closure result.

Proposition 5 *Let $P, Q \in \mathbf{3DP}$ and $\mu \in \omega(\mathcal{C}) \cup \rho(\mathcal{C})$. Then:*

1. $P \xrightarrow{\mu} Q$ implies $\text{shape}(Q) = \text{shape}(P)$;
2. P well-formed and $P \xrightarrow{\mu} Q$ implies Q well-formed.

5 Networks or 3D processes

Now we can define a network of 3D processes as a collection of 3D processes moving in the same 3D space.

Definition 18 (Networks of 3D processes) *The set \mathbb{N} of networks of 3D processes (3D networks, for short) is generated by the grammar:*

$$N ::= \text{Nil} \mid P \mid N \parallel N$$

where $P \in 3DP$. Given a finite set of indexes I , we often write $(\parallel P_i)_{i \in I}$ to denote the network that consists of all P_i with $i \in I$. We assume that $I = \emptyset$ implies $(\parallel P_i)_{i \in I} = \text{Nil}$. For $N = (\parallel P_i)_{i \in I}$ we let $S_i = \text{shape}(P_i)$, for $i \in I$, and define $\text{colliding}(N)$ as the set of all tuples $\langle P_i, P_j, X \rangle$ such that $\langle S_i, S_j, X \rangle \in \text{colliding}(\{S_i\}_{i \in I})$ (see Def. 8). A network $N = (\parallel P_i)_{i \in I}$ is said to be well-formed iff each P_i is well-formed and, for each pair of distinct processes P_i and P_j , the shapes S_i and S_j do not interpenetrate. Moreover, we extend the definition of **steer** on networks in the natural way, i.e. $\text{steer } t (\parallel P_i)_{i \in I} = (\parallel \text{steer } t P_i)_{i \in I}$.

In our running example we construct a network of processes containing a proper number of HEX, ATP and GLC processes in order to replicate the conditions in a portion of cytoplasm.

Definition 19 (Splitting bonds) *The function $\text{split} : 3DP \times \wp(\mathcal{C}) \rightarrow \mathbb{N}$ is defined as follows: If $\langle a, X \rangle \in \text{bonds}(P) \cap C$ and $P \equiv_P Q \langle a, X \rangle R$ then $\text{split}(P, C) = \text{split}(Q, C) \parallel \text{split}(R, C)$; if, otherwise, $\text{bonds}(P) \cap C = \emptyset$, then $\text{split}(P, C) = P$.*

It is worth noting that split shapes maintain the same velocity until the next occurrence of a movement time step. As we mentioned above, this is not a problem because they will not trigger a collision and, thus, a shorter first time of contact.

Proposition 6 *Let $P \in 3DP$ well-formed and $C \subseteq \mathcal{C}$. Then $\text{split}(P, C)$ is a well-formed network of 3D processes.*

Definition 20 (Semantics of weak- and strong-split operation) *Let $P \in 3DP$ a 3D process. If $P \searrow_{\mathcal{A}}$, we write that $P \xrightarrow{\rho} N \in \mathbb{N}$ iff there is a non empty set of channels $C = \{\langle a_1, X_1 \rangle, \dots, \langle a_n, X_n \rangle\} \subseteq \text{bonds}(P)$ such that $P = P_0 \xrightarrow{\rho(a_1, X_1)} P_1 \dots \xrightarrow{\rho(a_n, X_n)} P_n$, $P_n \not\rightarrow$ and $N = \text{split}(P_n, C)$. Similarly, $P \xrightarrow{\omega} N$ iff there is a channel $\langle a, X \rangle \in \text{bonds}(P)$ such that $P \xrightarrow{\omega(a, X)} Q$ and $N = \text{split}(Q, \{\langle a, X \rangle\})$.*

Since weak-split operations are due to a synchronization between just a pair of 3D processes, condition ‘ P is able to complete a reaction’ is not needed, but ‘ $P \xrightarrow{\omega(a, X)} Q$ ’ suffices to our aim.

Example 3 *Let us consider $P = H \langle \text{atp}, X \rangle (A \langle \text{glc}, Y \rangle G)$ where:*

$$H = S_h[\rho(\{\langle \text{atp}, X_{ha} \rangle, \langle \text{glc}, Y_{hg} \rangle\}).\text{HEX}],$$

$$A = S_a[\rho(\{\langle \overline{\text{atp}}, X_{ah} \rangle\}).\text{ADP} + \omega(\overline{\text{atp}}, X_{ah}).\text{ATP}],$$

$$G = S_g[\rho(\{\langle \overline{\text{glc}}, Y_{gh} \rangle\}).\text{G6P} + \omega(\overline{\text{glc}}, Y_{gh}).\text{GLC}],$$

$X'_{ha} \cap X'_{ah} = X$ (here X'_{ha} and X'_{ah} are the sites X_{ha} and X_{ah} expressed w.r.t. a global coordinate system; this convention will be applied later on) and $Y'_{hg} \cap Y'_{gh} = Y$. According to the definitions given so far, P is able to complete a reaction since:

$$\begin{aligned} P &\xrightarrow{\rho(\text{atp}, X)} S_h[\rho(\{\langle \text{glc}, Y_{hg} \rangle\}).\text{HEX}] \langle \text{atp}, X \rangle (S_a[\text{ADP}] \langle \text{glc}, Y \rangle G) \\ &\xrightarrow{\rho(\text{glc}, Y)} S_h[\text{HEX}] \langle \text{atp}, X \rangle (S_a[\text{ADP}] \langle \text{glc}, Y \rangle S_g[\text{G6P}]) = R \end{aligned}$$

Moreover, $R \not\rightarrow$ and

$$\begin{aligned} \text{split}(R, C) &= S_a[\text{HEX}] \parallel \text{split}(S_a[\text{ADP}] \langle \text{glc}, Y \rangle S_g[\text{G6P}], C) \\ &= S_h[\text{HEX}] \parallel (S_a[\text{ADP}] \parallel S_g[\text{G6P}]) = N \end{aligned}$$

where $C = \{\langle \text{atp}, X \rangle, \langle \text{glc}, Y \rangle\}$, implies $P \xrightarrow{\rho} N$. Moreover, let us note that, for each $t \in \mathbb{T}$, $P \xrightarrow{t} P$ but $P \not\xrightarrow{t}$ since P is able to complete a reaction and $P \not\rightarrow$ does not hold.

Below we define the temporal and functional behaviour of 3D networks. We assume that such networks perform basic actions that belong to set $\{\omega, \rho, \kappa\}$, where ω and ρ denote, resp., weak- and strong- split operations as a unique action (at the network level we only see whether shape bonds can be split or not) and κ represents system evolutions due to collision response (see Section 5.1). We also let elements of the set $\{\omega, \rho\} \cup \mathbb{T}$ to be ranged over by ν, ν', \dots .

Definition 21 (Temporal and Functional Behaviour of N-terms)

Rules in Table 5 defines the transition relations $\xrightarrow{t} \subseteq \mathbb{N} \times \mathbb{N}$ for $t \in \mathbb{T}$ and $\xrightarrow{\nu} \subseteq \mathbb{N} \times \mathbb{N}$ for $\nu \in \{\omega, \rho\}$. As usual, symmetric rules have been omitted.

A timed trace from N is a finite sequence of steps of the form $N = N_0 \xrightarrow{\nu_1} N_1 \cdots \xrightarrow{\nu_n} N_n = M$. We also write that $N \xrightarrow{t} M$ if there is a timed trace $N = N_0 \xrightarrow{\nu_1} N_1 \cdots \xrightarrow{\nu_n} N_n = M$ such that $t = \sum_{i=0}^n \{\nu_i \mid \nu_i \in \mathbb{T}\}$.

$\text{EMPTY}_t \frac{}{\text{Nil} \xrightarrow{t} \text{Nil}}$	$\text{PAR}_t \frac{N \xrightarrow{t} N' \quad M \xrightarrow{t} M'}{N \parallel M \xrightarrow{t} N' \parallel M'}$
$\text{PAR}_a \frac{N \xrightarrow{\nu} N'}{N \parallel M \xrightarrow{\nu} N' \parallel M}$	

Table 5: Temporal and functional behaviour of 3D networks

Proposition 7 *Let $t \in \mathbb{T}$, $P \in \mathbf{3DP}$, $N \in \mathbb{N}$, with P and N well-formed.*

1. $P \xrightarrow{\omega} N$ and $P \xrightarrow{\rho} N$ implies $N \in \mathbb{N}$ well-formed.
2. $N \xrightarrow{t} M$ implies M well-formed;
3. $N \xRightarrow{t} M$ implies M well-formed.

5.1 Collision response

In this section we describe the semantics of *collisions response*. As already said, the notion of compatibility between channels (and, hence, processes) has been introduced to distinguish between elastic and inelastic collision. In particular, collisions among compatible processes are always inelastic. So, if $P \xrightarrow{\langle a, X_p \rangle} P'$ and $Q \xrightarrow{\langle \bar{a}, X_q \rangle} Q'$, with $\langle a, X_p \rangle$ and $\langle \bar{a}, X_q \rangle$ compatible, and P and Q collide in the non-empty site $X = X_p \cap X_q$ we get a compound process $(P' \langle a, X \rangle Q') \llbracket \mathbf{v} \rrbracket$ where the velocity \mathbf{v} is provided by Equation (3) in Def. 9. Vice versa, a collision between two incompatible processes P and Q is treated as an elastic one. After such a collision, P and Q (actually the processes we obtain by updating their velocities according to Equations (1) and (2) in Def. 9) will proceed independently to each other.

To resolve collisions, we introduce two different kinds of *reduction relations* over 3D networks, namely $\xrightarrow{\langle P, Q, X \rangle}_e$ and $\xrightarrow{\langle P, Q, X \rangle}_i$, where P, Q are 3D processes and X is a surface of contact (see Table 6). Intuitively, if $N \xrightarrow{\langle P, Q, X \rangle}_e M$ ($N \xrightarrow{\langle P, Q, X \rangle}_i M$), then M is the 3D network we obtain once an elastic (inelastic, resp.) collision between P and Q in the surface of contact X has been resolved. These reduction relations also use the structural congruence over 3D networks.

Definition 22 (Structural congruence over 3D networks)

The structural congruence over terms in \mathbb{N} , that we denote with \equiv , is the smallest relation that satisfies the following axioms:

- $\text{Nil} \parallel N \equiv N$, $N \parallel M \equiv M \parallel N$ and $N \parallel (M \parallel R) \equiv (N \parallel M) \parallel R$;
- $P \parallel N \equiv Q \parallel N$ provided that $P \equiv_P Q$.

Rule ELAS in Table 6 simply changes velocities of two colliding but incompatible processes guided by Equations (1) and (2) in Def. 9, while rule INEL joins two compatible processes P and Q to obtain a compound process whose velocity is given by Equation (3) in Def. 9. Note that we force P and Q to synchronize on the execution of two compatible actions $\langle \alpha, X_p \rangle$ and $\langle \bar{\alpha}, X_q \rangle$ before joining them. In rules ELAS $_{\equiv}$ and INELAS $_{\equiv}$ we also consider structural congruence over nets of processes. In Def. 23 we collect together all the reduction-steps needed to solve collisions listed in a given set of collisions $\text{colliding}(N)$; clearly N is a generic 3D network.

Definition 23 (Resolving collisions) Let $N \in \mathbb{N}$ and $\langle P, Q, X \rangle$ a tuple in $\text{colliding}(N)$. $N \xrightarrow{\langle P, Q, X \rangle} M$ if either $P \sim Q$ and $N \xrightarrow{\langle P, Q, X \rangle}_i M$ or $P \not\sim Q$ and $N \xrightarrow{\langle P, Q, X \rangle}_e M$.

Moreover, we write that $N \xrightarrow{\kappa} M$ if either $\text{colliding}(N) = \emptyset$ and $N = M$ or $\text{colliding}(N) \neq \emptyset$ and there is a finite sequence of reduction steps $N = N_0 \xrightarrow{\langle P_1, Q_1, X_1 \rangle} N_1 \dots \xrightarrow{\langle P_k, Q_k, X_k \rangle} N_k = M$ such that:

1. $\langle P_i, Q_i, X_i \rangle \in \text{colliding}(N_{i-1})$ for each $i \in [1, k]$;
2. $\text{colliding}(N_k) = \emptyset$.

Let also note that, at any given time t , $\text{colliding}(N)$ can be obtained from the set of all the pairs of processes in N that are touching at that time. This set and hence $\text{colliding}(N)$ is surely finite and changes only when we resolve some inelastic collision (this is because, after an inelastic collision one or more binding sites can possibly become internal points of a compound process, and hence are not available anymore). Moreover a collision between pairs of processes with the same shape can not be resolved twice. This is either because two processes P and Q have been bond in a compound process as a consequence of an inelastic collision, or because P and Q collide elastically and their velocities have been changed according to Equations

$\text{ELAS} \frac{P \xleftrightarrow{X}_e Q = (\mathbf{v}_p, \mathbf{v}_q)}{(P \parallel Q) \parallel N \xrightarrow{\langle P, Q, X \rangle}_e (P[\mathbf{v}_p] \parallel Q[\mathbf{v}_q]) \parallel N}$	
$\text{ELAS} \equiv \frac{N \equiv N' \quad N' \xrightarrow{\langle P, Q, X \rangle}_e M}{N \xrightarrow{\langle P, Q, X \rangle}_e M}$	
$\text{INEL} \frac{P \xrightarrow{\langle \alpha, X_p \rangle} P' \quad Q \xrightarrow{\langle \bar{\alpha}, X_q \rangle} Q' \quad \alpha \in \{a, \bar{a}\} \quad P \xrightarrow{X_p \cap X_q}_i Q = \mathbf{v}}{(P \parallel Q) \parallel N \xrightarrow{\langle P, Q, X_p \cap X_q \rangle}_i ((P' \langle a, X_p \cap X_q \rangle Q')[\mathbf{v}]) \parallel N}$	
$\text{INEL} \equiv \frac{N \equiv N' \quad N' \xrightarrow{\langle P, Q, X \rangle}_i M}{N \xrightarrow{\langle P, Q, X \rangle}_i M}$	

Table 6: Reaction rules for elastic and inelastic collisions

(1) and (2) in Def. 9 in order to obtain two processes that do not collide anymore (see Lemma 5 in Appendix C). Thus, we can always decide if there is a finite sequence of reduction steps that allows us to resolve all collisions listed in $\text{colliding}(N)$ and hence obtain a network M with $\text{colliding}(M) = \emptyset$.

Proposition 8 *Let $N \in \mathbb{N}$, $P, Q \in 3\text{DP}$ and X a not-empty set subset of \mathbb{P} . Then N well-formed and $N \xrightarrow{\langle P, Q, X \rangle} M$ implies M well-formed.*

By iterative applications of Proposition 8 (see Appendix C for the proof) it is also:

Lemma 1 *Let N a well-formed 3D network. Then $N \xrightarrow{\kappa} M$ implies M well-formed.*

We are now ready to define how a network of 3D processes evolves by performing an infinite number of *movement time steps*.

Definition 24 (System evolution) *Let $N, M \in \mathbb{N}$ and $t, t' \in \mathbb{T}$. We say that $(N, t) \xRightarrow{t'} (M, t + t')$ iff one of the following conditions holds:*

1. $t' = \text{Ftoc}(N) \leq \Delta$ and $N \xRightarrow{t'} N' \xrightarrow{\kappa} N''$ and $M = \text{steer}(t + t') N''$;

2. $t' = \Delta < \text{Ftoc}(N)$ and $N \xRightarrow{t'} N'$ and $M = \text{steer}(t + t') N'$.

A system evolution is any infinite sequence of time steps of the form:

$$(N_0, 0) \xRightarrow{t_1} (N_1, t_1) \xRightarrow{t_2} (N_2, t_1 + t_2) \cdots (N_{i-1}, \sum_{j=1}^{i-1} t_j) \xRightarrow{t_i} (N_i, \sum_{j=1}^i t_j) \xRightarrow{t_{i+1}} \cdots$$

Note that, for each $i \geq 1$, $t_i = \min\{\text{Ftoc}(N_{i-1}), \Delta\}$ as discussed in Section 2. Moreover, in order to make sure that processes will never interpenetrate during a system evolution, if $\text{Ftoc}(N_{i-1}) \leq \Delta$, we first resolve all the collisions that happen after time $t_i = \text{Ftoc}(N_{i-1})$ (by means of transition $\xrightarrow{\kappa}$) and then apply the changes suggested by the function **steer** as described in Section 3.1.

Example 4 This example shows a possible evolution of the 3D network $N_0 = (\text{HEX} \parallel \text{ATP}) \parallel \text{GLC}$ where *HEX*, *ATP* and *GLC* are the 3D processes of Example 2. Below we use the following notation:

- $H(t) = (S_h + t)[\text{HEX}]$, $A(t) = (S_a + t)[\text{ATP}]$ and $G(t) = (S_g + t)[\text{GLC}]$ for each $t \in \mathbb{T}$. Note that $\text{HEX} = H(0)$, $\text{ATP} = A(0)$ and $\text{GLC} = G(0)$;
- $C = \rho(\{\langle \text{atp}, X_{ha} \rangle, \langle \text{glc}, X_{hg} \rangle\}).\text{HEX}$ and, for any $t \leq t_h$, $\text{HA}(t) = \omega(\text{atp}, X_{ha}).\text{HEX} + \epsilon(t_h - t). \langle \text{glc}, X_{hg} \rangle.C$;
- $\text{AH}(t) = \omega(\overline{\text{atp}}, X_{ah}).\text{ATP} + \epsilon(t_a - t). \rho(\{\overline{\text{atp}}, X_{ah}\}).\text{ADP}$ for any $t \in \mathbb{T}$ with $t \leq t_a$;
- $\text{GH}(t) = \omega(\overline{\text{glc}}, X_{gh}).\text{GLC} + \epsilon(t_g - t). \rho(\{\overline{\text{glc}}, X_{gh}\}).\text{G6P}$ for any $t \leq t_g$.

Let $t_1 = \text{Ftoc}(N_0)$ and assume $t_1 \leq \Delta$. By the operational rules, it is $N_0 \xRightarrow{t_1} H(t_1) \parallel A(t_1) \parallel G(t_1) = N'_0$. We also assume that $\text{colliding}(N'_0) = \{\langle H(t_1), A(t_1), X \rangle\}$ where $X = X'_{ha} \cap X'_{ah} \neq \emptyset$, $X'_{ha} = \text{global}(X_{ha}, \mathcal{R}(S_h + t_1))$ and $X'_{ah} = \text{global}(X_{ah}, \mathcal{R}(S_a + t_1))$. Then: $N'_0 \xrightarrow{\kappa} P(t_1) \parallel G(t_1) = N''_0$ where $P(t_1) = ((S_h + t_1)[\text{HA}(0)] \langle \text{atp}, X \rangle (S_a + t_1)[\text{AH}(0)]) \llbracket \mathbf{v}_{ha} \rrbracket$ and $\mathbf{v}_{ha} = H(t_1) \xleftrightarrow{X}_i A(t_1)$. Finally:

$$(N_0, 0) \xRightarrow{t_1} (N_1, t_1)$$

where $N_1 = \text{steer } t_1 N''_0 = \text{steer } t_1 P(t_1) \parallel \text{steer } t_1 G(t_1) = P(t_1) \llbracket \mathbf{v}_1 \rrbracket \parallel G(t_1) \llbracket \mathbf{v}_2 \rrbracket$. Note that:

$$\begin{aligned}
P(t_1)[\mathbf{v}_1] &= ((S_h + t_1)[\text{HA}(0)] \langle \text{atp}, X \rangle (S_a + t_1)[\text{AH}(0)])[\mathbf{v}_1] \\
&= (S_h^1[\text{HA}(0)] \langle \text{atp}, X \rangle S_a^1[\text{AH}(0)])
\end{aligned}$$

where $S_h^1 = ((S_h + t_1)[\mathbf{v}_1])$ and $S_a^1 = ((S_a + t_1)[\mathbf{v}_1])$. Moreover, $G(t_1)[\mathbf{v}_2] = ((S_g + t_1)[\mathbf{v}_2])[\text{GLC}] = S_g^1[\text{GLC}]$.

Let $t_2 = \text{Ftoc}(N_1)$ and assume $t_2 = t_h \leq \min\{t_a, \Delta\}$ ¹³. Below we write $G'(t_2)$ and $P'(t_2)$ to denote, respectively, the 3D processes $(S_g^1 + t_2)[\text{GLC}]$ and $((S_h^1 + t_2)[\text{HA}(t_h)] \langle \text{atp}, X + t_2 \cdot \mathbf{v}_1 \rangle (S_a^1 + t_2)[\text{AH}(t_2)])$. Again by the operational rules, $N_1 \xrightarrow{t_2} P'(t_2) \parallel G'(t_2) = N_1'$.

Let $\text{colliding}(N_1') = \{ \langle P'(t_2), G'(t_2), Y \rangle \}$ where $Y = X'_{hg} \cap X'_{gh} \neq \emptyset$, $X'_{hg} = \text{global}(X_{hg}, \mathcal{R}(S_h^1 + t_2))$ and $X'_{gh} = \text{global}(X_{gh}, \mathcal{R}(S_g^1 + t_2)) \subseteq \mathcal{B}(P'(t_2))$. If $\mathbf{v}_{gh} = P'(t_2) \xleftarrow{Y}_i G'(t_2)$, then $N_1' \xrightarrow{t_2} (P(t_2) \langle \text{glc}, Y \rangle G(t_2))[\mathbf{v}_{gh}] = N_1''$ where $G(t_2) = (S_g^1 + t_2)[\text{GH}(0)]$ and $P(t_2) = ((S_h^1 + t_2)[\text{C}] \langle \text{atp}, X + t_2 \cdot \mathbf{v}_1 \rangle (S_a^1 + t_2)[\text{AH}(t_2)])$. Finally:

$$(N_1, t_1) \xrightarrow{t_2} (N_2, t_1 + t_2)$$

where $N_2 = \text{steer}(t_1 + t_2) N_1'' = (P(t_2) \langle \text{glc}, Y \rangle G(t_2))[\mathbf{v}_3]$. Observe that:

$$\begin{aligned}
(P(t_2) \langle \text{glc}, Y \rangle G(t_2))[\mathbf{v}_3] &= (P(t_2)[\mathbf{v}_3]) \langle \text{glc}, Y \rangle (G(t_2)[\mathbf{v}_3]) = \\
&= ((S_h^2[\text{C}] \langle \text{atp}, X + t_2 \cdot \mathbf{v}_1 \rangle S_a^2[\text{AH}(t_2)]) \langle \text{glc}, Y \rangle S_g^2[\text{GH}(0)])[\mathbf{v}_3]
\end{aligned}$$

where $S_h^2 = (S_h^1 + t_2)[\mathbf{v}_3]$, $S_a^2 = (S_a^1 + t_2)[\mathbf{v}_3]$ and $S_g^2 = (S_g^1 + t_2)[\mathbf{v}_3]$.

At this stage the network contains just one process and, as a consequence, no collisions are possible. Thus, $\text{Ftoc}(N_2) = \infty$. Assume $t_g = t_a - t_2 \leq \Delta$ ¹⁴. If we let $X_g = (X + t_2 \cdot \mathbf{v}_1) + t_g \cdot \mathbf{v}_3$ and $Y_g = Y + t_g \cdot \mathbf{v}_3$, then:

$$\begin{aligned}
N_2 &\xrightarrow{t_g} ((S_h^2 + t_g)[\text{C}] \langle \text{atp}, X_g \rangle (S_a^2 + t_g)[\text{AH}(t_a)]) \langle \text{glc}, Y_g \rangle (S_g^2 + t_g)[\text{GH}(t_g)] \xrightarrow{\rho} \\
&= ((S_h^2 + t_g)[\text{HEX}] \parallel (S_a^2 + t_g)[\text{ADP}]) \parallel (S_g^2 + t_g)[\text{G6P}] \xrightarrow{\Delta - t_g} \\
&= ((S_h^2 + \Delta)[\text{HEX}] \parallel (S_a^2 + \Delta)[\text{ADP}]) \parallel (S_g^2 + \Delta)[\text{G6P}] = N_2'. \text{ Thus:}
\end{aligned}$$

$$(N_2, t_1 + t_2) \xrightarrow{\Delta} (N_3, t_1 + t_2 + \Delta)$$

¹³If were $t_2 < t_h$, the 3D processes $\text{HA}(t_2)$ and GLC would be no more compatible, and a collision between them would be treated as elastic. On the other hand, if were $t_2 = t_a$ the idling time for $\text{AH}(t_a)$ would be over. As a consequence, time would pass further only after the execution of a weak-split operation that splits the bond between the Hexokinases and the Atp molecules

¹⁴If were $t_g \neq t_a - t_2$ the reaction could never proceed since the involved molecules would never be able to release – all together – the bonds. Thus the system would deadlock.

where $N_3 = \text{steer } (t_1 + t_2 + \Delta) \ N'_2 = (S_h^3[\text{HEX}] \parallel S_a^3[\text{ADP}]) \parallel S_g^3[\text{G6P}]$,
 $S_i^3 = (S_i^2 + \Delta) \llbracket \mathbf{v}_i \rrbracket$ and $\mathbf{v}_i = \text{steer } (t_1 + t_2 + \Delta) \ (S_i^2 + \Delta)$ for each $i \in \{h, a, g\}$.

We can prove the following basic property of the Shape Calculus stating that any system evolution does not introduce space inconsistencies like interpenetration of 3D processes or not well-formed processes.

Theorem 1 (Closure w.r.t. well-formedness) *Let N be a well-formed network of 3D processes. If $(N, t) \xRightarrow{t'} (M, t + t')$ then M is well-formed.*

Proof: Assume that $(N, t) \xRightarrow{t'} (M, t + t')$ because of $N \xRightarrow{t'} N' \xrightarrow{\kappa} N''$ and $M = \text{steer } (t + t') \ N''$ (the other case – see Def. 24 – is similar). Then, by Proposition 7 (item 3) and Lemma 1, N well-formed and $N \xRightarrow{t'} N' \xrightarrow{\kappa} N''$ implies N'' and hence $M = \text{steer } (t + t') \ N''$ well-formed.

6 Conclusions and Future Work

We have defined the full timed operational semantics of the Shape Calculus and we have introduced a notion of well-formedness of the different objects of the calculus. We proved that the evolution of a well-formed network of 3D processes is always well-formed, that is to say, no spatial or temporal inconsistencies can be introduced by the dynamics of the calculus. As future work we intend to provide verification techniques for the Shape Calculus. In order to do this we believe that a sort of “tailoring” should be made on the calculus, making some parts (e.g. movement) more abstract and other parts (e.g. behaviours) more specific adding quantitative information (for instance probabilities or costs). The whole process will then be supported by the definition of proper logical languages to specify properties of interests. Of course we expect that some approximations will be necessary to reach computability and/or feasibility. Another direction of future work is the possibility to include in the calculus some new useful, but in some cases complex, concepts such as re-shaping, message passing of values, and communication by perception of a compatible process in the neighbourhood.

References

- [1] <http://www.rcsb.org>, RCSB - Protein Data Bank.

- [2] R. Barbuti, A. Maggiolo-Schettini, P. Milazzo, and G. Pardini. Spatial Calculus of Looping Sequences. *Electr. Notes Theor. Comput. Sci.*, 229(1):21–39, 2009. FBTC 2008.
- [3] R. Barbuti, A. Maggiolo-Schettini, P. Milazzo, G. Pardini, and L. Tesei. Spatial P Systems. *Natural Computing*, 2010. Received: 26 October 2009 Accepted: 24 February 2010 Published online: 24 March 2010.
- [4] E. Bartocci, D. R. Cacciagrano, M. R. Di Berardini, E. Merelli, and L. Tesei. Timed Operational Semantics and Well-formedness of Shape Calculus. *Scientific Annals of Computer Science*, 20, 2010.
- [5] E. Bartocci, F. Corradini, M. R. Di Berardini, E. Merelli, and L. Tesei. Shape Calculus. A spatial mobile calculus for 3D shapes. *Scientific Annals of Computer Science*, 20, 2010.
- [6] E. Bartocci, M. R. Di Berardini, F. Corradini, M. Emanuela, and L. Tesei. A Shape Calculus for Biological Processes. In *ICTCS 2009*, pages 30–33, 2009.
- [7] L. Bortolussi and A. Policriti. Stochastic concurrent constraint programming and differential equations. *Electr. Notes Theor. Comput. Sci.*, 190(3):27–42, 2007.
- [8] F. Buti, D. Cacciagrano, F. Corradini, E. Merelli, M. Pani, and L. Tesei. Bone remodelling in BioShape. In *CS2BIO 2010: Proc. of Interactions between Computer Science and Biology, 1st International Workshop*, 2010. Avail. at <http://cosy.cs.unicam.it/bioshape/cs2bio2010.pdf>.
- [9] F. Buti, D. Cacciagrano, F. Corradini, E. Merelli, and L. Tesei. BioShape: a spatial shape-based scale-independent simulation environment for biological systems. In *ICCS 2010: Proc. of Simulation of Multiphysics Multiscale Systems, 7th International Workshop*, 2010. Avail. at <http://cosy.cs.unicam.it/bioshape/iccs2010.pdf>.
- [10] D. Cacciagrano, F. Corradini, and M. Merelli. Bone remodelling: a Complex Automata-based model running in BioShape. In *ACRI 2010: Proc. of Cellular Automata for Research and Industry, 9th International Conference*, 2010. Avail. at <http://cosy.cs.unicam.it/bioshape/acri2010.pdf>.

- [11] L. Cardelli. Brane calculi. In *Proc. of CMSB '04*, volume 3082 of *LNCS*, pages 257–278, 2004.
- [12] F. Ciocchetta and J. Hillston. Bio-pepa: An extension of the process algebra pepa for biochemical networks. *Electr. Notes Theor. Comput. Sci.*, 194(3):103–117, 2008.
- [13] C. Ericson. *Real-time collision detection*. Elsevier North-Holland, Inc., 2005.
- [14] A. Finkelstein, J. Hetherington, L. Li, O. Margoninski, P. Saffrey, R. Seymour, and A. Warner. Computational challenges of systems biology. *IEEE Computer*, 37(5):26–33, 2004.
- [15] C. Hecker. Physics, part 3: Collision response. *Game Developer Magazine*, pages 11–18, 1997.
- [16] M. John, R. Ewald, and A. Uhrmacher. A spatial extension to the π calculus. *Electronic Notes in Theoretical Computer Science*, 194(3):133–148, 2008. FBTC 2007.
- [17] R. Milner. *Communication and concurrency*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1989.
- [18] C. Priami and P. Quaglia. Beta binders for biological interactions. In *CMSB*, pages 20–33, 2004.
- [19] A. Regev, E. Panina, W. Silverman, L. Cardelli, and E. Shapiro. Bioambients: an abstraction for biological compartments. *Theoretical Computer Science*, 325(1):141–167, 2004.
- [20] BIOSHAPE. <http://cosy.cs.unicam.it/bioshape/>.
- [21] W. Yi. Real-time behaviour of asynchronous agents. In *CONCUR*, pages 502–520, 1990.

Appendix A: Proofs of Section 3

The following lemmas are first needed.

Lemma 2 *Let $S \in \mathbb{S}$ well-formed and $t \in \mathbb{T}$. Then:*

1. $\mathbf{v}(S+t) = \mathbf{v}(S)$ and $m(S+t) = m(S)$;
2. $\mathcal{R}(S+t) = \mathcal{R}(S) + t \cdot \mathbf{v}(S)$;
3. $\mathcal{P}(S+t) = \mathcal{P}(S) + t \cdot \mathbf{v}(S)$ (and, hence, $\mathcal{B}(S+t) = \mathcal{B}(S) + t \cdot \mathbf{v}(S)$).

Proof: We prove Items 1, 2 and 3 by induction on $S \in \mathbb{S}$ well-formed.

Basic: $S = \langle V, m, \mathbf{p}, \mathbf{v} \rangle$, $S+t = \langle V + (t \cdot \mathbf{v}), m, \mathbf{p} + (t \cdot \mathbf{v}), \mathbf{v} \rangle$ and:

1. $\mathbf{v}(S+t) = \mathbf{v} = \mathbf{v}(S)$ and $m(S+t) = m = m(S)$;
2. $\mathcal{R}(S+t) = \mathbf{p} + (t \cdot \mathbf{v}) = \mathcal{R}(S) + (t \cdot \mathbf{v})$;
3. $\mathcal{P}(S+t) = V + (t \cdot \mathbf{v}(S)) = \mathcal{P}(S) + (t \cdot \mathbf{v}(S))$.

Comp: $S = S_1 \langle X \rangle S_2$ and $S+t = (S_1+t) \langle Y \rangle (S_2+t)$ where $Y = X + t \cdot \mathbf{v}$ and $\mathbf{v} = \mathbf{v}(S) = \mathbf{v}(S_i)$. By induction hypothesis:

1. $\mathbf{v}(S_i+t) = \mathbf{v}(S_i) = \mathbf{v}$ and $m(S_i+t) = m(S_i)$ for $i = 1, 2$. Thus: $\mathbf{v}(S+t) = \mathbf{v}$ and $m(S+t) = m(S_1+t) + m(S_2+t) = m(S_1) + m(S_2) = m(S)$.

$$2. \mathcal{R}(S+t) = \frac{\sum_{i=1}^2 m(S_i+t) \cdot \mathcal{R}(S_i+t)}{\sum_{i=1}^2 m(S_i+t)} = \frac{\sum_{i=1}^2 m(S_i) \cdot (\mathcal{R}(S_i) + t \cdot \mathbf{v})}{\sum_{i=1}^2 m(S_i)} = \frac{\sum_{i=1}^2 m(S_i) \cdot \mathcal{R}(S_i)}{\sum_{i=1}^2 m(S_i)} + t \cdot \mathbf{v} = \mathcal{R}(S) + t \cdot \mathbf{v}$$

3. $\mathcal{P}(S+t) = \mathcal{P}(S_1+t) \cup \mathcal{P}(S_2+t) = \mathcal{P}(S_1) \cup \mathcal{P}(S_2) = \mathcal{P}(S)$.

To prove Prop. 1 we also need the following lemma; its proof has been omitted because it is similar to that of Lemma 2.

Lemma 3 *Let $S \in \mathbb{S}$ well-formed and $\mathbf{w} \in \mathbb{V}$. Then:*

1. $\mathbf{v}(S[\mathbf{w}]) = \{\mathbf{w}\}$ and $m(S[\mathbf{w}]) = m(S)$;
2. $\mathcal{P}(S[\mathbf{w}]) = \mathcal{P}(S)$ (and, hence, $\mathcal{B}(S[\mathbf{w}]) = \mathcal{B}(S)$).

Proposition 1 *Let $S \in \mathbb{S}$, $t \in \mathbb{T}$ and $\mathbf{w} \in \mathbb{V}$. If S is well-formed then $S+t$ and $S[\mathbf{w}]$ are well-formed.*

Proof: We prove these statement by induction in $S \in \mathbb{S}$.

Basic: $S = \langle V, m, \mathbf{p}, \mathbf{v} \rangle$. Both $S+t = \langle V + t \cdot \mathbf{v}, m, \mathbf{p} + t \cdot \mathbf{v}, \mathbf{v} \rangle$ and $S[\mathbf{w}] = \langle V, m, \mathbf{p}, \mathbf{w} \rangle$ are well-formed shapes.

Comp: $S = S_1 \langle X \rangle S_2$. Let $\mathbf{v} = \mathbf{v}(S) = \mathbf{v}(S_i)$ for $i = 1, 2$. By Defs 5 and 6, it is $S+t = (S_1+t) \langle Y \rangle (S_2+t)$ where $Y = X + t \cdot \mathbf{v}$, and $S[\mathbf{w}] = (S_1[\mathbf{w}]) \langle X \rangle (S_2[\mathbf{w}])$. We first prove $S+t$ wellformedness.

1. By induction hypothesis, $S_1 + t$ and $S_2 + t$ are well-formed;
2. S well-formed implies that X is non-empty set that is equal to $\mathcal{P}(S_1) \cap \mathcal{P}(S_2) = \mathcal{B}(S_1) \cap \mathcal{B}(S_2)$. By Lemma 2, we also have that $Y = X + t \cdot \mathbf{v}$ is non-empty and $Y = \mathcal{P}(S_1 + t) \cap \mathcal{P}(S_2 + t) = \mathcal{B}(S_1 + t) \cap \mathcal{B}(S_2 + t)$.
3. By Lemma 2-1, $\mathbf{v}(S + t) = \mathbf{v}$ where $\mathbf{v} = \mathbf{v}(S_i + t) = \mathbf{v}(S_i)$ for $i = 1, 2$.

Similarly we can prove that $S[\mathbf{w}]$ is well-formed. Shapes $S_1[\mathbf{w}]$ and $S_2[\mathbf{w}]$ are well-formed (this follows by induction hypothesis); by Lemma 3-1 it is $\mathbf{v}(S[\mathbf{w}]) = \mathbf{w}$; finally $\emptyset \neq X = \mathcal{P}(S_1) \cap \mathcal{P}(S_2) = \mathcal{B}(S_1) \cap \mathcal{B}(S_2)$, $\mathcal{P}(S_i) = \mathcal{P}(S_i[\mathbf{w}])$ and $\mathcal{B}(S_i) = \mathcal{B}(S_i[\mathbf{w}])$ for $i = 1, 2$ (by Lemma 3-2) imply $\emptyset \neq X = \mathcal{P}(S_1[\mathbf{w}]) \cap \mathcal{P}(S_2[\mathbf{w}]) = \mathcal{B}(S_1[\mathbf{w}]) \cap \mathcal{B}(S_2[\mathbf{w}])$.

Appendix B: Proofs of Section 4

This section is devoted to prove main results stated in Section 4.

Proposition 3 *Let $P \in \text{3DP}$ well-formed. Either $P \xrightarrow{t} Q$ or $P \xrightarrow{\mu} Q$ implies $Q \in \text{3DP}$ well-formed.*

Proof: We first prove (by induction on $P \in \text{3DP}$) that:

1. $P \xrightarrow{t} Q$ implies $\text{shape}(Q) = \text{shape}(P) + t$;
2. $P \xrightarrow{\mu} Q$ implies $\text{shape}(Q) = \text{shape}(P)$.

Basic: $P = S[B]$.

1. By operational rules, $P \xrightarrow{t} Q$ implies $B \xrightarrow{t} B'$ and $Q = (S + t)[B']$. Thus: $\text{shape}(Q) = S + t = \text{shape}(P) + t$.
2. $P \xrightarrow{\mu} Q$ implies $Q = S[B']$ for a proper $B' \in \mathbb{B}$ (see rule BASIC_c in Table 3). Hence $\text{shape}(Q) = S = \text{shape}(P)$.

Comp: $P = P_1 \langle a, X \rangle P_2$.

1. $P \xrightarrow{t} Q$ implies $P_1 \xrightarrow{t} Q_1$, $P_2 \xrightarrow{t} Q_2$, and $Q = Q_1 \langle a, Y \rangle Q_2$ where $Y = X + (t \cdot \mathbf{v}(P))$. By induction hypothesis, $\text{shape}(Q) = \text{shape}(Q_1) \langle Y \rangle \text{shape}(Q_2) = (\text{shape}(P_1) + t) \langle Y \rangle (\text{shape}(P_2) + t) = \text{shape}(P) + t$.
2. $P \xrightarrow{\mu} Q$ implies either $P_1 \xrightarrow{\mu} Q_1$ and $Q = Q_1 \langle a, X \rangle P_2$ or $P_2 \xrightarrow{\mu} Q_2$ and $Q = P_1 \langle a, X \rangle Q_2$. Let us consider the former case (the latter one can be proved similarly). By induction hypothesis it is $\text{shape}(Q) = \text{shape}(Q_1) \langle X \rangle \text{shape}(P_2) = \text{shape}(P_1) \langle X \rangle \text{shape}(P_2) = \text{shape}(P)$.

Now we prove that P well-formed and $P \xrightarrow{t} Q$ imply Q well-formed. Again, we proceed by induction on P well-formed.

Basic: $P = S[B]$. If $P \xrightarrow{t} Q$ then $Q = (S + t)[B']$ where $B \xrightarrow{t} B'$. In this case, S well-formed and Prop. 1 imply $\text{shape}(Q) = S + t$ well-formed. Moreover $\text{global}(X, \mathcal{R}(P)) \subseteq \mathcal{B}(P)$ for each site $X \subseteq \mathbb{P}$ that occurs in B and, hence, in B' . By Lemma 2-4, $\text{global}(X, \mathcal{R}(Q)) = \text{global}(X, \mathcal{R}(P) + t \cdot \mathbf{v}(P)) = \text{global}(X, \mathcal{R}(P)) + t \cdot \mathbf{v}(P) \subseteq \mathcal{B}(P) + t \cdot \mathbf{v}(P) = \mathcal{B}(Q)$. Thus, Q is well-formed.

Comp: $P = P_1 \langle a, X \rangle P_2$ with P_i well-formed for $i = 1, 2$ and $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$. $P \xrightarrow{t} Q$ implies $P_i \xrightarrow{t} Q_i$, for $i = 1, 2$, and $Q = Q_1 \langle a, Y \rangle Q_2$ with $Y = X + (t \cdot \mathbf{v}(P))$. By induction hypothesis, Q_1 and Q_2 are well-formed. Moreover, $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$ implies $Y = X + t \cdot \mathbf{v}(P) \subseteq (\mathcal{B}(P_1) \cap \mathcal{B}(P_2)) + t \cdot \mathbf{v}(P) = (\mathcal{B}(P_1) + t \cdot \mathbf{v}(P)) \cap (\mathcal{B}(P_2) + t \cdot \mathbf{v}(P)) = \mathcal{B}(Q_1) \cap \mathcal{B}(Q_2)$.

It remains to prove, by induction on P , that P well-formed and $P \xrightarrow{\mu} Q$ imply Q well-formed.

Basic: $P = S[B]$. If $P \xrightarrow{\mu} Q$ then $B \xrightarrow{\mu} B'$ for a proper $B' \in 3\text{DP}$ and $Q = S[B']$. P well-formed implies S well-formed. Moreover, for each X that occurs in B' (and hence in B), $\text{global}(X, \mathcal{R}(P)) \subseteq \mathcal{B}(P) = \mathcal{B}(S) = \mathcal{B}(Q)$. So, Q is well-formed.

Comp: $P = P_1 \langle a, X \rangle P_2$ with P_i well-formed for $i = 1, 2$ and $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$. $P \xrightarrow{\mu} Q$ implies either $P_1 \xrightarrow{\mu} Q_1$ and $Q = Q_1 \langle a, X \rangle P_2$ or $P_2 \xrightarrow{\mu} Q_2$ and $Q = P_1 \langle a, X \rangle Q_2$. We only prove the former case (the latter one is similar). By induction hypothesis Q_1 is well-formed; moreover, $\text{shape}(Q_1) = \text{shape}(P_1)$ and, hence, $\mathcal{B}(Q_1) = \mathcal{B}(P_1)$. Thus, Q_1 and P_2 are well-formed with $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2) = \mathcal{B}(Q_1) \cap \mathcal{B}(P_2)$, i.e. Q is well-formed.

Proposition 4 *Let $P \in 3\text{DP}$ well-formed. If $\langle a, X \rangle \in \text{bonds}(P)$ there is a well-formed 3D process $Q \langle a, X \rangle R \equiv_P P$.*

Proof: By induction on P well-formed.

Basic: $P = S[B]$. This case is not possible since $\text{bonds}(P) = \emptyset$.

Comp: $P = P_1 \langle b, Y \rangle P_2$ with P_i well-formed for $i = 1, 2$, $Y \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$ and $\mathbf{v}(P_1) = \mathbf{v}(P_2) = \mathbf{v}(P) = \mathbf{v}$. If $\langle a, X \rangle = \langle b, Y \rangle$ it suffices to choose $Q = P_1$ and $R = P_2$. Assume $\langle a, X \rangle \in \text{bonds}(P_1)$ (the case in which $\langle a, X \rangle \in \text{bonds}(P_2)$ is symmetric). By induction hypothesis, $Q_1 \langle a, X \rangle R_1 \equiv_P P_1$ is well-formed and $P = P_1 \langle b, Y \rangle P_2 \equiv_P (Q_1 \langle a, X \rangle R_1) \langle b, Y \rangle P_2$. We have two possible subcases:

1. $Y \subseteq \mathcal{B}(R_1)$ and $P \equiv_P Q \langle a, X \rangle R$ where $Q = Q_1$ and $R = R_1 \langle b, Y \rangle P_2$,
 2. $Y \subseteq \mathcal{B}(Q_1)$ and $P \equiv_P Q \langle a, X \rangle R$ where $Q = R_1$ and $R = Q_1 \langle b, Y \rangle P_2$.
- We only consider the former case (the latter one is similar) and prove that

$Q \langle a, X \rangle R$ is well-formed. To this aim we have to show that: (i) $R = R_1 \langle b, Y \rangle P_2$ is well-formed (note that $P_1 \equiv_P Q_1 \langle a, X \rangle R_1$ well-formed and $Q = Q_1$ implies Q_1 well-formed), (ii) $\mathbf{v}(Q) = \mathbf{v}(R)$ (this follows easily because P well-formed implies $\mathbf{v}(Q_1)$ (i.e. $\mathbf{v}(Q)) = \mathbf{v}(R_1) = \mathbf{v}(P_2) = \mathbf{v}(R)$) and (iii) $X \subseteq \mathcal{B}(Q) \cap \mathcal{B}(R)$. Below we prove first (i) and then (iii)

$P_1 \equiv_P Q_1 \langle a, X \rangle R_1$ and $P = P_1 \langle b, Y \rangle P_2$ well-formed imply, resp., R_1 and P_2 well-formed; moreover, $\mathbf{v}(R_1) = \mathbf{v}(P_1) = \mathbf{v} = \mathbf{v}(P_2)$ (because $P_1 \equiv_P Q_1 \langle a, X \rangle R_1$ is well-formed). To show that $R = R_1 \langle b, Y \rangle P_2$ is well-formed, it remains to prove that $Y \subseteq \mathcal{B}(R_1) \cap \mathcal{B}(P_2)$. We proceed by contradiction. Assume $\mathbf{y} \in Y$ such that $\mathbf{y} \notin \mathcal{B}(R_1) \cap \mathcal{B}(P_2)$. Then, $Y \subseteq \mathcal{B}(R_1)$ implies $\mathbf{y} \notin \mathcal{B}(P_2)$ and, hence, $\mathbf{y} \notin \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$. But, this is impossible because $Y \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$.

We prove that $X \subseteq \mathcal{B}(Q) \cap \mathcal{B}(R) = \mathcal{B}(Q_1) \cap \mathcal{B}(R)$ by contradiction. Recall that $Q_1 \langle a, X \rangle R_1$ well-formed implies $X \subseteq \mathcal{B}(Q_1) \cap \mathcal{B}(R_1)$. Assume that there is $\mathbf{x} \in X \subseteq \mathcal{B}(Q_1) \cap \mathcal{B}(R_1)$ such that $\mathbf{x} \notin \mathcal{B}(Q_1) \cap \mathcal{B}(R)$. Then $\mathbf{x} \notin \mathcal{B}(R) = (\mathcal{B}(R_1) \cup \mathcal{B}(P_2)) \setminus \{\mathbf{z} \mid \mathbf{z} \text{ is interior of } \mathcal{P}(R_1) \cup \mathcal{P}(P_2)\}$. Moreover, $\mathbf{x} \in X \subseteq \mathcal{B}(R_1) \subseteq \mathcal{B}(R_1) \cap \mathcal{B}(P_2)$ and $\mathbf{x} \notin \mathcal{B}(R)$ implies that \mathbf{x} is an interior point of $\mathcal{P}(R_1) \cup \mathcal{P}(P_2)$, i.e. \mathbf{x} is an interior point of $\mathcal{P}(P_2)$; this is because $\mathbf{x} \in \mathcal{B}(R_1)$ implies that \mathbf{x} cannot be an interior point of $\mathcal{P}(R_1)$. Finally, $\mathbf{x} \in X$ implies that \mathbf{x} is also an interior point of $P_1 \equiv_P Q_1 \langle a, X \rangle R_1$. Thus, P_1 and P_2 interpenetrate each other. But this is not possible since $P \equiv_P P_1 \langle b, Y \rangle P_2$ is well-formed.

Proposition 5 *Let $P, Q \in 3DP$ and $\mu \in \omega(\mathcal{C}) \cup \rho(\mathcal{C})$. Then:*

1. $P \xrightarrow{\mu} Q$ implies $\text{shape}(Q) = \text{shape}(P)$;
2. P well-formed and $P \xrightarrow{\mu} Q$ implies Q well-formed.

Proof: We only prove the statement for $\mu = \rho(a, X)$; if $\mu = \omega(a, X)$ the statement can be proved similarly.

Basic: $P = S[B]$. This case is not possible since $P \not\xrightarrow{\rho(a, X)}$.

Comp: $P = P_1 \langle b, Y \rangle P_2$ with P_i well-formed for $i = 1, 2$ and $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$. We distinguish two possible subcases:

(i) $\langle b, Y \rangle = \langle a, X \rangle$, $P_1 \xrightarrow{\rho(\alpha, X_1)} Q_1$, $P_2 \xrightarrow{\rho(\bar{\alpha}, X_2)} Q_2$ with $\alpha \in \{a, \bar{a}\}$ and $X = X_1 \cap X_2$, and $Q = Q_1 \langle a, X \rangle Q_2$.

1. By Prop. 3-2, $\text{shape}(Q_i) = \text{shape}(P_i)$ for $i = 1, 2$ and, hence, $\text{shape}(Q) = \text{shape}(Q_1) \langle X \rangle \text{shape}(Q_2) = \text{shape}(P_1) \langle X \rangle \text{shape}(P_2) = \text{shape}(P)$.

2. By induction hypothesis, P_i well-formed implies Q_i well-formed, for $i = 1, 2$. Moreover $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2)$ and, again, $\text{shape}(Q_i) = \text{shape}(P_i)$ for $i = 1, 2$ imply $X \subseteq \mathcal{B}(Q_1) \cap \mathcal{B}(Q_2)$. So the 3D process Q is well-formed.

(ii) Either $P_1 \xrightarrow{\rho(a,X)} Q_1$ and $Q = Q_1 \langle a, X \rangle P_2$ or $P_2 \xrightarrow{\rho(a,X)} Q_2$ and $Q = P_1 \langle a, X \rangle Q_2$.

1. By induction hypothesis it is $\text{shape}(Q_i) = \text{shape}(P_i)$ for $i = 1, 2$. As in the previous case, we can prove that $\text{shape}(Q) = \text{shape}(P)$.

2. By induction hypothesis we have that Q_i and P_j , for $\{i, j\} \in \{1, 2\}$, are well-formed. Moreover, by Item 1, $\text{shape}(Q_i) = \text{shape}(P_i)$, and hence $X \subseteq \mathcal{B}(P_1) \cap \mathcal{B}(P_2) = \mathcal{B}(Q_1) \cap \mathcal{B}(Q_2)$. Also in this case we can conclude that Q is well-formed.

Appendix C: Proofs of Section 5

Proposition 6 *Let $P \in \text{3DP}$ well-formed and $C \subseteq \mathcal{C}$. Then $\text{split}(P, C)$ is a well-formed network of 3D processes .*

Proof: By induction on the number of channels in $\text{bonds}(P) \cap C$.

$\text{bonds}(P) \cap C = \emptyset$. In such a case $\text{split}(P, C) = P$ is well-formed.

$\langle a, X \rangle \in \text{bonds}(P) \cap C \neq \emptyset$. By Prop. 4, there are P_1, P_2 well-formed such that $P \equiv_P P_1 \langle a, X \rangle P_2$ and $\text{split}(P, C) = \text{split}(P_1, C) \parallel \text{split}(P_2, C) = N_1 \parallel N_2$. By induction hypothesis N_i is a well-formed 3D network for $i = 1, 2$. Now, in order to prove that $\text{split}(P, C)$ is well-formed, it still remains to prove that if Q_1 and Q_2 are two 3D processes in the network N_1 and N_2 , respectively, then Q_1 and Q_2 do not interpenetrate. Assume, towards a contradiction, that there are Q_1 and Q_2 in N_1 and N_2 , respectively, such that Q_1 and Q_2 interpenetrate, i.e. such that there is (at least) a point \mathbf{x} that is interior of both Q_1 and Q_2 .

For each $i = 1, 2$, it holds that either $\text{bonds}(P_i) \cap C = \emptyset$, $N_i = P_i$ and $Q_i = P_i$ or $\text{bonds}(P_i) \cap C = \{ \langle a_1, X_1 \rangle, \langle a_2, X_2 \rangle, \dots, \langle a_{ni}, X_{ni} \rangle \}$, $P_i \equiv_P P_i^1 \langle a_1, X_1 \rangle (P_i^2 \langle a_2, X_2 \rangle \dots (P_i^{ni} \langle a_{ni}, X_{ni} \rangle P_i^{ni+1}))$ (this follows by iterating Prop. 4), $N_i = P_i^1 \parallel (P_i^2 \parallel \dots (P_i^{ni} \parallel P_i^{ni+1}))$ and $Q_i = P_i^j$ for some $j = 1, 2, \dots, ni, ni + 1$. Thus, in both cases the set of interior points of Q_i is a subset of interior point of P_i . As a consequence, if Q_1 and Q_2 interpenetrate so do P_1 and P_2 . But this is not possible because P is well-formed. To prove Prop. 7 we need the following preliminary result.

Lemma 4 *Let $N, M \in \mathbb{N}$, $P, Q \in 3DP$ and $t \in \mathbb{T}$ such that $N \xrightarrow{t} M$ and $P \xrightarrow{t} Q$. The network N contains P iff M contains Q .*

Proof: By induction on $N \in \mathbb{N}$.

$N = \text{Nil}$. $N \xrightarrow{t} \text{Nil}$ and Nil does not contain any 3D process.

$N = P$ with $P \in 3DP$. By our operational semantics, $N \xrightarrow{t} M$ iff $M = Q$ and $P \xrightarrow{t} Q$.

$N = N_1 \parallel N_2$. $N \xrightarrow{t} M_1 \parallel M_2 = M$ iff $N_i \xrightarrow{t} M_i$ for $i = 1, 2$. P is contained in N iff P is contained either in N_1 or in N_2 iff (by induction hypothesis) and Q is contained either in M_1 or in M_2 , i.e. iff Q is contained in M .

Proposition 7 *Let $t \in \mathbb{T}$, $P \in 3DP$, $N \in \mathbb{N}$, with P and N well-formed.*

1. $P \xrightarrow{\omega} N$ and $P \xrightarrow{\rho} N$ implies $N \in \mathbb{N}$ well-formed.
2. $N \xrightarrow{t} M$ implies M well-formed;
3. $N \Rightarrow M$ implies M well-formed.

Proof: Item 1 can be proved by (iterative applications of) Prop. 5-2 and Prop. 6. Item 3 follows directly from Items 1 and 2 (see Definition 21). In what follows, we prove Item 2 by induction on $N \in \mathbb{N}$.

$N = \text{Nil}$. In such a case $N \xrightarrow{t} M$ implies $M = \text{Nil}$ that is well-formed.

$N = P$ with $P \in 3DP$. The statement is a trivial consequence of Prop. 3-3.

$N = N_1 \parallel N_2$. If $N \xrightarrow{t} M$ then $N_i \xrightarrow{t} M_i$, for $i = 1, 2$, and $M = M_1 \parallel M_2$. By induction hypothesis, M_1 and M_2 are well-formed. So, it remains to prove that if Q_1 and Q_2 are 3D processes that compose the networks M_1 and M_2 , resp., then Q_1 and Q_2 do not interpenetrate. We proceed by contradiction. Assume that there are Q_1 in M_1 and Q_2 in M_2 that interpenetrate each other. By Lemma 4, M_i contains Q_i iff N_i contains P_i for some $P_i \in 3DP$ such that $P_i \xrightarrow{t} Q_i$. Moreover, by Prop. 3-1, $\text{shape}(Q_i) = \text{shape}(P_i) + t$. Thus, if Q_1 and Q_2 interpenetrate the same do P_1 and P_2 . But this not possible since N is well-formed.

Lemma 5 *Let $N, M \in \mathbb{N}$ such that N is well-formed and $N \xrightarrow{\langle P, Q, X \rangle} M$ for some $\langle P, Q, X \rangle \in \text{colliding}(N)$. If P' and Q' are two processes composing M with $\text{shape}(P') = \text{shape}(P)$ and $\text{shape}(Q') = \text{shape}(Q)$, then $\langle P', Q', Y \rangle \notin \text{colliding}(M)$ for any $Y \subseteq \mathbb{P}$.*

Proof: By rules in Table 6, $N \xrightarrow{\langle P, Q, X \rangle}_e M$ implies $N \xrightarrow{\langle P, Q, X \rangle}_e M$ implies $M \equiv (P \llbracket \mathbf{v}_p \rrbracket \parallel Q \llbracket \mathbf{v}_q \rrbracket) \parallel N'$ where the pair of velocity $(\mathbf{v}_p, \mathbf{v}_q) = P \xleftarrow{X}_e Q$. Moreover, Equations (1) and (2) in Def. 9 ensure that $P \llbracket \mathbf{v}_p \rrbracket$ and $Q \llbracket \mathbf{v}_q \rrbracket$ (since N is well-formed, these are the only processes in M with the same shape of P and Q , resp.) can not collide anymore. Now assume $N \xrightarrow{\langle P, Q, X \rangle}_i M$. By Def. 23, there are compatible $\langle \alpha, X_p \rangle, \langle \bar{\alpha}, X_q \rangle \in \mathcal{C}$ such that $P \xrightarrow{\langle \alpha, X_p \rangle} P'$, $Q \xrightarrow{\langle \bar{\alpha}, X_q \rangle} Q'$ and $M \equiv ((P' \langle a, X \rangle Q') \llbracket \mathbf{v} \rrbracket) \parallel N'$ with $\alpha \in \{a, \bar{a}\}$, $X = X_p \cap X_q \neq \emptyset$ and $\mathbf{v} = P \xleftarrow{X}_i Q$. The statement follows because there are no P' and Q' in M such that $\text{shape}(P') = \text{shape}(P)$ and $\text{shape}(Q') = \text{shape}(Q)$.

Proposition 8 *Let $N \in \mathbb{N}$, $P, Q \in \text{3DP}$ and X a not-empty set subset of \mathbb{P} . Then N well-formed and $N \xrightarrow{\langle P, Q, X \rangle}_e M$ implies M well-formed.*

Proof: If $N \xrightarrow{\langle P, Q, X \rangle}_e M$ then $N \equiv (P \parallel Q) \parallel N'$, $M \equiv (P \llbracket \mathbf{v}_p \rrbracket \parallel Q \llbracket \mathbf{v}_q \rrbracket) \parallel N'$ and $(\mathbf{v}_p, \mathbf{v}_q) = P \xleftarrow{X}_e Q$. Moreover, N well-formed implies that P , Q and N' are well-formed and processes P and Q do not interpenetrate each other and with any other 3D process composing N' (see Def. 18). Finally: $P \llbracket \mathbf{v}_p \rrbracket$ and $Q \llbracket \mathbf{v}_q \rrbracket$ are well-formed 3D processes with $\text{shape}(P \llbracket \mathbf{v}_p \rrbracket) = \text{shape}(P)$ and $\text{shape}(Q \llbracket \mathbf{v}_q \rrbracket) = \text{shape}(Q)$ (see Lemma 3). Thus, $P \llbracket \mathbf{v}_p \rrbracket$ and $Q \llbracket \mathbf{v}_q \rrbracket$ do not interpenetrate each other and with any other 3D process in N' and we can conclude that M is well-formed.

Now assume $N \xrightarrow{\langle P, Q, X \rangle}_i M$. Again, $N \equiv (P \parallel Q) \parallel N'$ and, by rule INEL, there are $\langle \alpha, X_p \rangle, \langle \bar{\alpha}, X_q \rangle \in \mathcal{C}$ compatible such that $\alpha \in \{a, \bar{a}\}$, $P \xrightarrow{\langle \alpha, X_p \rangle} P'$, $Q \xrightarrow{\langle \bar{\alpha}, X_q \rangle} Q'$ and $M \equiv ((P' \langle a, X \rangle Q') \llbracket \mathbf{v} \rrbracket) \parallel N'$ where $X = X_p \cap X_q \neq \emptyset$ and $\mathbf{v} = P \xleftarrow{X}_i Q$. By Prop. 3-3, P and Q well-formed implies P' and Q' well-formed. Moreover:

(1) $P \xrightarrow{\langle \alpha, X_p \rangle} P'$ and $Q \xrightarrow{\langle \bar{\alpha}, X_q \rangle} Q'$ imply $X_p \subseteq \mathcal{B}(P)$ and $X_q \subseteq \mathcal{B}(Q)$, resp. Thus, $X = X_p \cap X_q \subseteq \mathcal{B}(P) \cap \mathcal{B}(Q) = \mathcal{B}(P') \cap \mathcal{B}(Q')$ (since Prop. 3-2 implies $\text{shape}(P') = \text{shape}(P)$ and $\text{shape}(Q') = \text{shape}(Q)$), $P' \langle a, X \rangle Q'$ and $(P' \langle a, X \rangle Q') \llbracket \mathbf{v} \rrbracket$ are well-formed 3D processes.

(2) $\text{shape}((P' \langle a, X \rangle Q') \llbracket \mathbf{v} \rrbracket) = \text{shape}(P' \langle a, X \rangle Q') = \mathcal{P}(P') \cup \mathcal{P}(Q') = \mathcal{P}(P) \cup \mathcal{P}(Q)$. As a consequence, $(P' \langle a, X \rangle Q') \llbracket \mathbf{v} \rrbracket$ can not interpenetrate any 3D process that compose the network N' (otherwise either P or Q must do the same).

Also in this case we can conclude that M is well-formed.